# Multi-Scale Ensemble for Road Segmentation using Super-Resolution

**(Group: Sleeplearning)**

Marios Dimitriadis
ETH Zurich
mdimitriadis@student.ethz.ch

Adam Klebus
ETH Zurich
klebusa@student.ethz.ch

Guy Shacht
ETH Zurich
gshacht@student.ethz.ch

Matas Udris
ETH Zurich
mudris@student.ethz.ch

## Abstract

*In recent years, deep learning methods have surpassed traditional methods in many computer-vision tasks. Two notable examples are semantic segmentation and super-resolution. However, state-of-the-art semantic segmentation methods perform better when applied to input images of high resolution, which can be hard to obtain.*

*To this end, we introduce a method leveraging a deep super-resolution method to further enhance state-of-the-art semantic segmentation methods. We do this by allowing those methods to work on super-resolved inputs, which are of higher quality and richer level of detail.*

*Furthermore, we leverage the variance obtained by working on multiple input scales by forming an ensemble method combining multiscale predictions. We show that segmentation methods are more accurate on those super-resolved inputs, and that our ensemble method surpasses each individual method and achieves a score of 0.92644 on the Kaggle challenge.*

## 1. Introduction and Related Works

We begin with an introductory section explaining the foundations of our method. Our method combines modern approaches to semantic segmentation, super-resolution and ensemble methods. Therefore, we start by laying out a brief explanation of each of those fields and discuss their recent developments.

### 1.1. Semantic Segmentation

Semantic Segmentation is the task of identifying and distinguishing objects within a provided image with the goal of classifying each pixel. Classical approaches for semantic segmentation were based on clustering, for example, grouping nearby pixels by the similarity of their color [10]. Such methods have been since surpassed by modern deep-learning-based methods, particularly CNNs. Two prominent examples are U-Net [17] and DeepLabV3 [3]. The U-Net is a decoder-encoder-like architecture, using an equal number of downsampling and upsampling layers as well as skip connections to provide context to upsampling layers [17]. DeepLabV3 uses a backbone CNN to extract features and atrous convolutions [21] to preserve spacial information while reducing resolution [3]. DeepLabV3Plus extends this architecture by adding a decoder module for refinement of results [5]. U-Net and DeepLabV3(Plus) and their variants had achieved state-of-the-art results in image segmentation at the time of their introduction [3, 5, 17], and have been successfully applied in aerial image road segmentation [2, 8, 12, 22].

### 1.2. Super-Resolution

Super-resolution aims to transform a lower resolution input image into an enhanced, higher resolution version. Naive interpolation does not enhance the clarity of the image, thus initial attempts used an example-based approach [9], aiming to find similarities between the input images and their large, super-resolved counterparts. Recent approaches leverage deep learning. Dong et al. have introduced the deep method SRCNN [7]. Their proposed model combined global, super-resolution features with local, spatial information features, to produce the output. Such approaches were shown to surpass example-based methods. A notable improvement upon SRCNN is the VDSR network [13]. Instead of upscaling the input, the network takes an already naively upscaled image and predicts only the missing enhancement in a residual fashion, which was shown to improve accuracy [13].

Developments in SR have also improved semantic segmentation methods. Wang et al. introduced the DSRL, a framework that can be appended to existing architectures

such as DeepLabV3Plus, improving their accuracy [19]. DSRL guides the segmentation model to learn to upscale the input image and uses this to improve the mapping from encoder features to the final mask [19].

Another important work is the attention mechanism for weighting multi-scale features proposed by Chen et al. [4], shown to improve upon methods like DeepLabV3. Multi-scale features are attained by applying super-resolution methods and are locally combined using an attention mechanism. In our approach, we take inspiration from this idea.

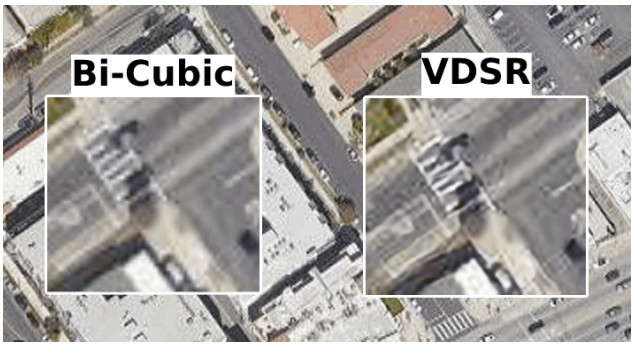


Figure 1. Our method leverages the state-of-the-art super-resolution method VDSR [13]. In the figure, a small patch taken from the $800 \times 800$ super-resolved satellite image is enlarged and compared to the naive bi-cubic upsampling method. By allowing segmentation methods to work on such high-resolution inputs, we obtain segmentation mask predictions of higher accuracy. One example is given in Figure 4.

### 1.3. Ensemble Methods

Ensemble methods combine predictions from various models to achieve greater performance than any of the individual components. The component predictions are typically combined through mean, median or majority voting, depending on the task [23]. Ensembles require a diverse set of base methods which can, in particular, be achieved through varying model hyper-parameters and input data [16, 18, 23]. Diverse methods produce independent errors which can be cancelled out when combined. Ensembles have been employed for the task of semantic road segmentation [1, 6].

## 2. Model and methods overview

wIn developing our final model, we went through a series of baseline designs, which are introduced in this section. Models with a 256 suffix appended to their names were trained on $256 \times 256$ crops from the original dataset, and models with a 512 suffix were trained on super-resolved crops of size $512 \times 512$ as explained in Algorithm 8. All encoders were pretrained on ImageNet. Henceforth, we denote the composition of a $k \times k$ convolution layer, a batch normalization layer, and a ReLU layer as a $k \times k$ CBR block. Our final model is an ensemble of 4 models, combining multiple scales and designs, and is denoted **U-Lab-MS** (**M**ulti**S**cale).

### 2.1. U-Net256

This was chosen as our first baseline, as it is a robust model that performs well in most cases. Its small size seems to be an advantage in preventing overfitting on small datasets. There are 5 skip connections between the encoder and the decoder. The first 4 are identity mappings, whereas the deepest one is a bridge consisting of two $3 \times 3$ CBR blocks. At each skip connection point, the decoder path feature map undergoes an upsampling operation and is concatenated with the skip connection feature map. The concatenated feature map undergoes two successive $3 \times 3$ CBR blocks. The upsampling module and the two $3 \times 3$ CBR blocks between two successive skip connection points are a U-Net decoder block.

### 2.2. DeepLabV3Plus256

This model has a very different architecture to U-Net; U-Net relies on 5 encoder-decoder skip connections, whereas DeepLabV3Plus only has two such connections: a simple $3 \times 3$ separable CBR block and a deep encoder-decoder bridge, the ASPP. The former operates on a high-resolution feature map of the encoder, whereas the latter operates on the deepest, lowest resolution one. The ASPP consists of a special pooling operation, 3 dilated convolutions, and a $1 \times 1$ CBR block. The special pooling operation consists of a global average pooling layer (which reduces each channel to a scalar), followed by a $1 \times 1$ CBR block and a bilinear upsampling that restores the original resolution. The dilated convolutions are $3 \times 3$ CBR blocks, with dilation rates of 12, 24, and 36. All 5 output volumes are stacked together and then forwarded to a final $1 \times 1$ CBR block. Due to the lack of an extensive decoder such as in U-Net, the output resolution of DeepLabV3Plus is 4 times smaller than the network input. Thus, the network has to undergo a naive bilinear upsampling of factor 4, which produces crude results.

### 2.3. U-Net512 and DeepLabV3Plus512

The difference in architecture between U-Net and DeepLabV3Plus indicates a potentially high variance between the predicted masks of the models, which is useful for ensembling.

Our first two baselines were trained on $256 \times 256$ crops. We then decided to use $512 \times 512$ crops as well in order to train our segmentation models on more detailed inputs. Thus, we pretrained the super-resolution network VDSR on the BSD500 [14] dataset, and used it to perform super-resolution on each $256 \times 256$ crop to obtain $512 \times 512$ crops.


Figure 2. Our final model, U-Lab-MS, ensembles together 4 different models, which are diverse. Therefore, prediction errors from each individual model are likely to cancel one another, so that the averaged mask is better than any individual mask.

During inference, we use the same VDSR network to transform the full $400 \times 400$ images to $800 \times 800$ images. The exact training algorithm is given in 8.

The performance of the 512-based methods is given in 3. One visual comparison is given in 4.

### 2.4. U-Net-Edge256

Inspired by papers like HED-Unet (Heidler et al. [11]), we coupled each satellite input with a corresponding on-the-fly computed edge map, extracted VGG19 features from the edge map, fused the resulting features to satellite VGG19 features using concatenation, and gated the fused features to the decoder part of U-Net. The hope was that due to the commonality between road masks and edge maps, the U-Net network can use the edge map encoded features as priors and produce better segmentation masks. We experimented with two types of edge maps, one computed by a pretrained network from HED, [20] [15] and the other with OpenCV's Canny() function with a threshold of 300. An example of a computed edge map is given in Figure 3.

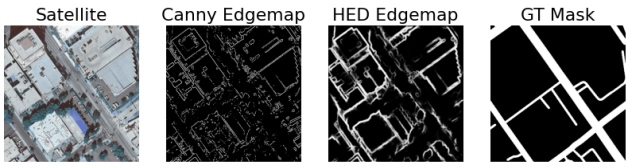


Figure 3. Despite the apparent commonality between road detection and edge map prediction, we did not manage to improve the baseline model using guidance from edge maps (both Canny and HED).

### 2.5. U-NetASPP512

To address the lack of a parameterized final layer in DeepLabV3Plus, we take inspiration from the U-Net and combine the two models to produce the U-NetASPP, which is a DeepLabV3Plus model with the final upsampling layer removed. Instead, we concatenate a skip connection from the early (high resolution) encoder layers with the decoder path and forward both into a U-Net decoder block. The resulting resolution is only 2 times smaller than the input, and then we apply a bilinear interpolation with a scale of 2 instead of 4.

The dilation rates used as default in the standard library implementation of DeepLabV3Plus are not the ones stipulated by the original paper, so we address this issue by replacing [12,24,36] with [6,12,18] ([12,24,36] are still valid for different settings) [5]. The intuitive reason for this choice is that the previous rates are too large, and the dilated convolutions will then mostly be applied on the zero-padding volume, which means the $3 \times 3$ convolution degenerates to $1 \times 1$ and does not achieve the purpose of dilation: to aggregate global information.

### 2.6. Final solution: U-Lab-MS

Our best-performing model is an ensemble method. It aims to leverage the diversity obtained by working on multiple scales. It is a 4-way ensemble between the four models **U-Net256**, **U-Net512**, **DeepLabV3Plus256** and **DeepLabV3Plus512**.

A simple average was found to perform best in combining the masks from all methods. The way we combined predicted $512 \times 512$ masks with $256 \times 256$ masks was to simply downscale the larger masks with a scale factor of 2 before averaging. We used OpenCV's resize function with INTER_AREA interpolation. The performance is given in 3. One visual comparison which shows the ensemble method is better than any base method is given in 2.

### 2.7. Augmentations and Training Details

In order to overcome the small size of the dataset and prevent overfitting, a data augmentation pipeline was selected through trial and error. It consists of a rotation by a uniformly sampled angle, a random square cropping, and a horizontal flip.

For training VDSR we used the Adam optimizer with default settings, a learning rate of $0.001$, batch size of 32, and MSE loss. For all segmentation network training, we used the Adam optimizer with a learning rate of $0.0001$ for 200 epochs and default settings. We used a batch size of 4 and the dice loss function, which improves performance for unbalanced classes, such as the distribution of roads in satellite images. We evaluated models before submission by using an $85\%$ training-validation split on the original dataset. All

**Algorithm 1:** Training of a $512 \times 512$ based network $N_\theta$, such as DeepLabV3Plus512 or U-Net512

**Data:** Training set $D = \{(x_i, y_i)\}_{i \in \{1,2,...\}}$
**Output:** Segmentation network $N_\theta$

Train VDSR network $V$ `// we use the SR dataset BSD300 and a scale factor of 2`
**while** *not yet converged* **do**
  Sample $(x, y)$ from $D$ randomly
  $x_{\text{large}} = \text{V}(x)$ `// the resolution of x is increased from 400 to 800`
  $y_{\text{large}} = \text{opencv.resize}(y, \text{scale=2}))$
  $\hat{x}, \hat{y} = \text{augment}(x_{\text{large}}, y_{\text{large}})$ `// we take a random patch of size 512, rotate and flip it randomly`
  $y_{\text{pred}} = N_\theta(\hat{x})$
  $\theta = \text{backpropogate}(\theta, \text{soft\_dice\_loss}(y_{\text{pred}}, \hat{y}))$

models were initialized with ImageNet-pretrained encoder weights except for one ablation study model that investigates the effect of pretraining.

To attain a full-size mask during inference from the $256 \times 256$ or $512 \times 512$ segmentation network outputs, we slide a window over the full image, apply the network to each window, and average pixels which appear in more than one window.

## 3. Results and Ablation studies

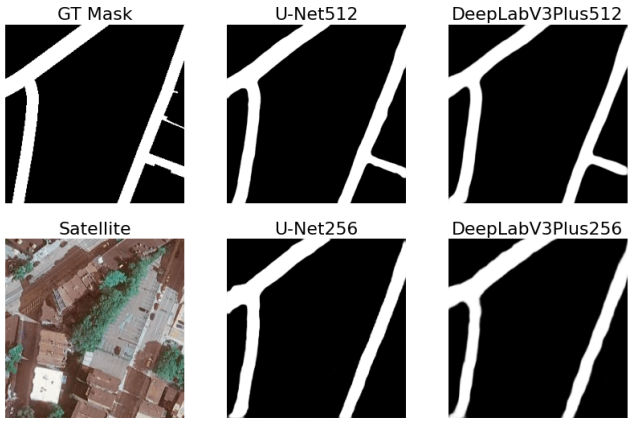


Figure 4. In this example, only U-Net512 and DeepLabV3Plus512, which are trained on super-resolved crops, are able to produce a particular road segment. An example of the gap in input quality was given in Figure 1

.

To evaluate the effects of each component, we performed extensive ablation experiments with many different combinations of settings. Augmentation and pretraining were important. As expected, augmentation alleviated overfitting

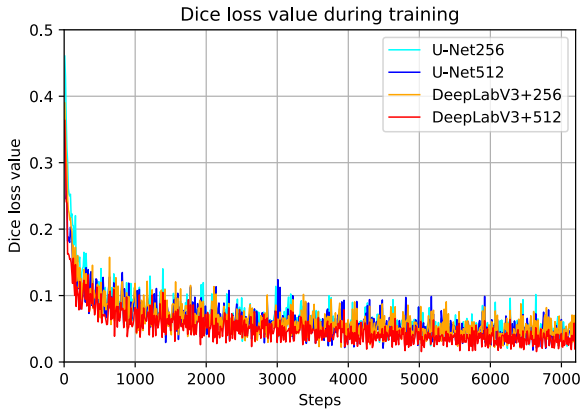


Figure 5. The training Dice loss of four of our baseline models over 200 epochs.

| Method | Train Dice | Train F1 | Kaggle F1 |
|---|---|---|---|
| U-Net256 (-A, -P) | 0.0139 | 0.960 | 0.864 |
| U-Net256 (-A) | 0.0109 | 0.968 | 0.898 |
| U-Net256 (-P) | 0.0652 | 0.787 | 0.897 |
| U-Net256 | 0.0484 | 0.844 | 0.919 |
| U-Net-ASPP512 | 0.0340 | 0.905 | 0.916 |
| DeepLabV3Plus256 | 0.0414 | 0.879 | 0.916 |
| U-Net512 | 0.0392 | 0.886 | 0.919 |
| DeepLabV3Plus512 | 0.0435 | 0.882 | 0.920 |
| U-Net-Edge | 0.0579 | 0.892 | 0.913 |
| U-Lab256 | 0.1442 | 0.855 | **0.921** |
| U-Lab512 | 0.1295 | 0.870 | **0.924** |
| U-Lab-MS | 0.1286 | 0.871 | **0.926** |

Table 1. Ablation studies. -A = no augmentation, -P = no encoder pretraining. Both are used for all models unless stated otherwise.

significantly, increasing the Kaggle F1 score. Pretraining the encoder showed a similar improvement in the Kaggle F1 score. Combining both techniques produced even better results.

U-NetASPP improved on the training metrics but regressed on the Kaggle score, indicating that the added parameters led to overfitting. U-Net-Edge was similar. Ensembling U-Net with DeepLabV3Plus (U-Lab) always outperformed individual models, and the 4-way ensemble, which also combines two scales, resulted in enough variance to create the best performing ensemble, U-Lab-MS.

## 4. Code

The implementation of our method is found at http://www.github.com/camillarhodes/CIL_Sleeplearning under the main branch.

## References


[1] Jose M Alvarez, Yann LeCun, Theo Gevers, and Antonio M Lopez. Semantic road segmentation via multi-scale ensembles of learned features. In *European Conference on Computer Vision*, pages 586–595. Springer, 2012. 2

[2] Alexander V. Buslaev, Selim S. Seferbekov, Vladimir I. Iglovikov, and Alexey A. Shvets. Fully convolutional network for automatic road extraction from satellite imagery. *CoRR*, abs/1806.05182, 2018. 1

[3] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation, 2017. 1

[4] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3640–3649, 2016. 2

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation, 2018. 1, 3

[6] Ahana Roy Choudhury, Biswas Parajuli, and Piyush Kumar. Quadroad: an ensemble of cnns for road segmentation. *Procedia Computer Science*, 176:138–147, 2020. 2

[7] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks, 2015. 1

[8] Shouji Du, Shihong Du, Bo Liu, and Xiuyuan Zhang. Incorporating deeplabv3+ and object-based image analysis for semantic segmentation of very high resolution remote sensing images. *International Journal of Digital Earth*, 14, 10 2020. 1

[9] William T. Freeman, Thouis R. Jones, and Egon C. Pasztor. Example-based super-resolution, Aug 2001. 1

[10] Yanming Guo, Yu Liu, Theodoros Georgiou, and Michael S. Lew. A review of semantic segmentation using deep neural networks. *International Journal of Multimedia Information Retrieval*, 7(2):87–93, Jun 2018. 1

[11] Konrad Heidler, Lichao Mou, Celia Baumhoer, Andreas Dietz, and Xiao Xiang Zhu. Hed-unet: Combined segmentation and edge detection for monitoring the antarctic coastline. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–14, 2021. 3

[12] Corentin Henry, Seyed Majid Azimi, and Nina Merkle. Road segmentation in SAR satellite images with deep fully convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 15(12):1867–1871, dec 2018. 1

[13] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks, 2015. 1, 2

[14] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001. 2

[15] Simon Niklaus. A reimplementation of HED using PyTorch. https://github.com/sniklaus/pytorch-hed, 2018. 3

[16] David Opitz and Richard Maclin. Popular ensemble methods: An empirical study. *Journal of artificial intelligence research*, 11:169–198, 1999. 2

[17] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 1

[18] Robert E Schapire. The strength of weak learnability. *Machine learning*, 5(2):197–227, 1990. 2

[19] Li Wang, Dong Li, Yousong Zhu, Lu Tian, and Yi Shan. Dual super-resolution learning for semantic segmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3773–3782, 2020. 2

[20] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *IEEE International Conference on Computer Vision*, 2015. 3

[21] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions, 2015. 1

[22] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. Road extraction by deep residual u-net. *CoRR*, abs/1711.10684, 2017. 1

[23] Zhi-Hua Zhou. *Ensemble Methods: Foundations and Algorithms*. Chapman and Hall/CRC, 2012. 2