Vision and Cognitive Systems

Project

Multi-scale patch-wise semantic segmentation of satellite images using U-Net architecture

Authors

Eugenio Fella - Theivan Pasupathipillai - Carlo Sgorlon Gaiatto

Introduction

Semantic segmentation of satellite images: automatic interpretation of large-scale geographic information.

Possible applications include road extraction, building detection and land cover classification (useful for sustainable development, agriculture, forestry and urban planning).

The state-of-the-art in this field is driven by architectures based on CNNs. In our project we presented a pipeline designed specifically for remote sensing data.

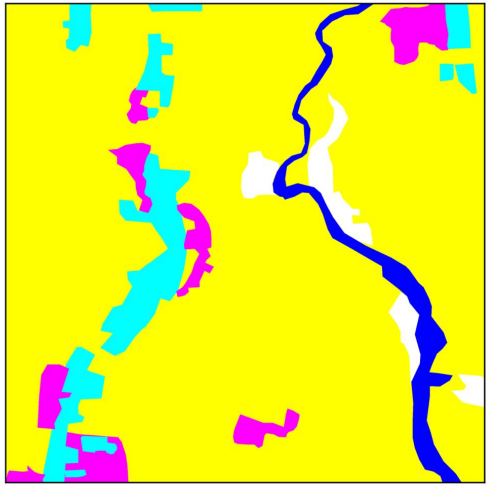
Dataset: Features

We analyzed the dataset from **DeepGlobe Land Cover Classification challenge** (CVPR 2018) consisting of 803 RGB satellite images focusing on rural areas.

Each image has a size of 2448x2448 pixels (0.5 meters per pixel) and it is paired with a mask for land cover annotation.

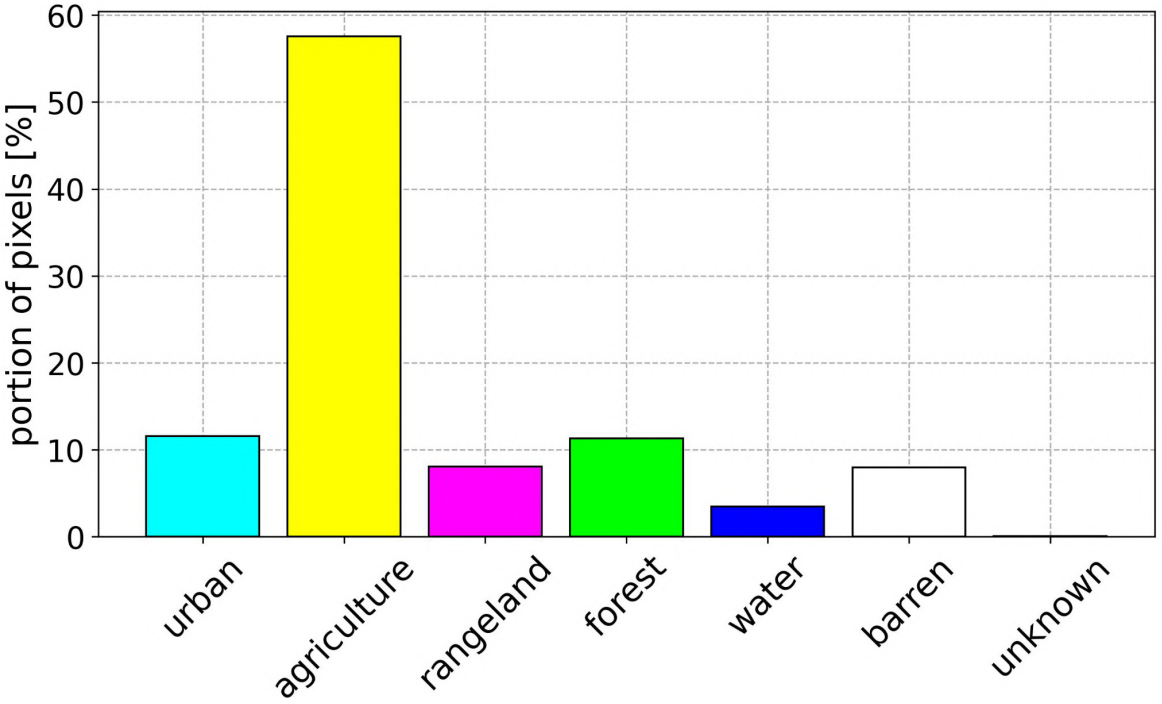
Image

Mask



Dataset: Classes

The dataset has **7 classes**, encoded using RGB values, and highly unbalanced:



Method

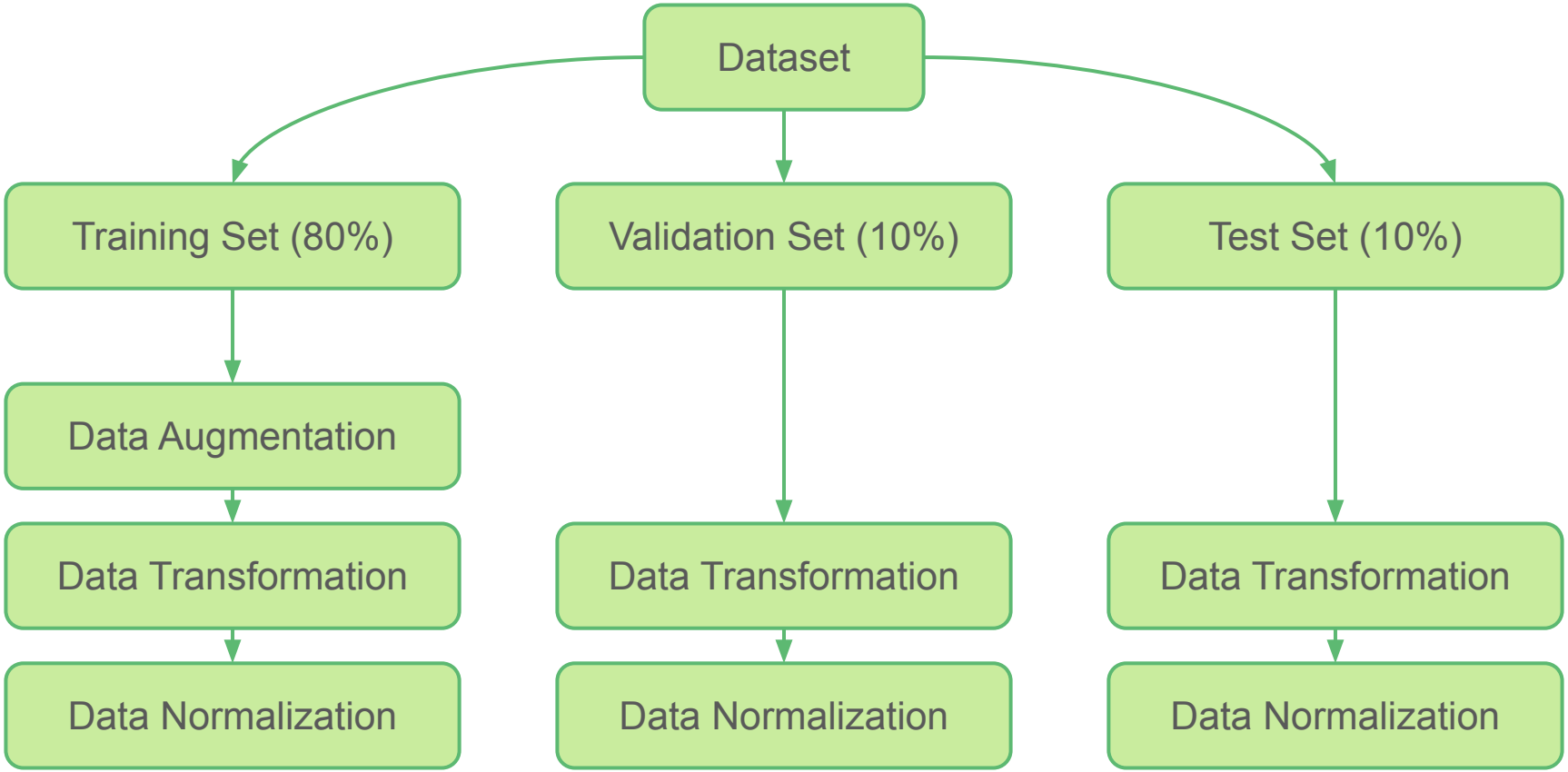
We splitted the dataset into **training, validation and test sets**, each with 642, 80 and 81 images (80%/10%/10%).

We defined a **pre-processing pipeline** that includes several data augmentation and data transformation techniques, taking into account the peculiar characteristics of satellite imagery.

Then, we applied z-score normalization per channel using mean and standard deviation computed across the entire training set (feature-wise pixel standardization).

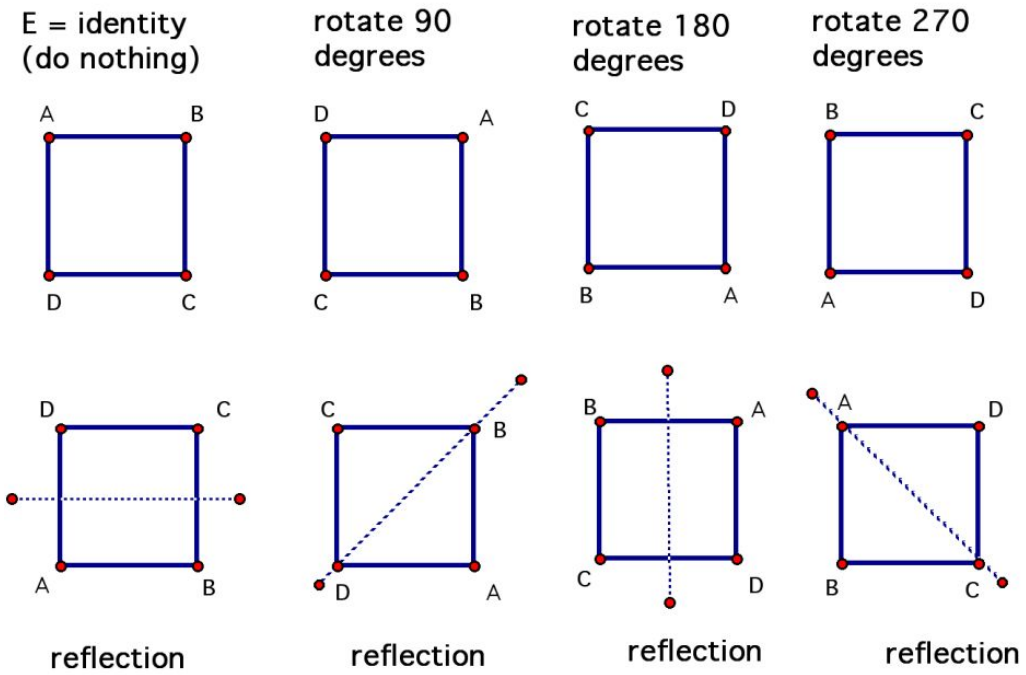
Note: the models have been trained using an NVIDIA Tesla T4 GPU with 16GB of memory, running on a pre-built virtual machine in Google Cloud.

Method: Scheme



Data Augmentation: Dihedral Group D4

Random rotation of 0/90/180/270 degrees, horizontal/vertical flip and transpositions.



Data Augmentation: Rescaling

Rescale with a randomly selected factor between 0.5 and 1.25.

scale = 100 %

scale = 10 %

Data Augmentation: Color jittering

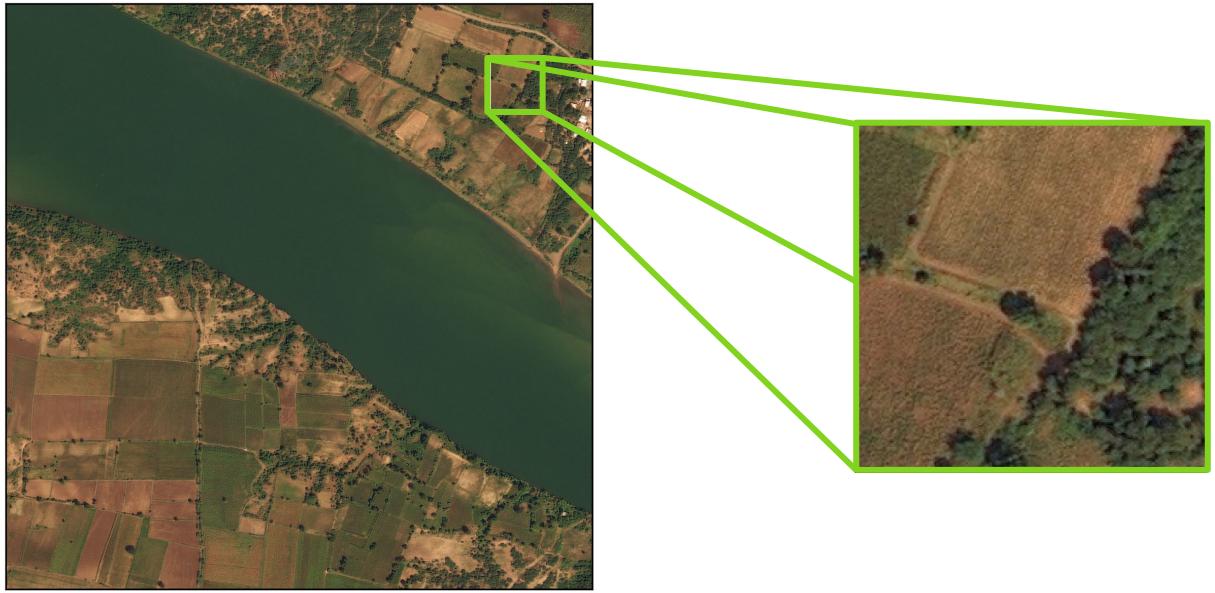
Random change in brightness, contrast and saturation.

color jitter = Off

color jitter = On

Data Transformation: Cropping

Random cropping of a patch of 224x224 pixels.



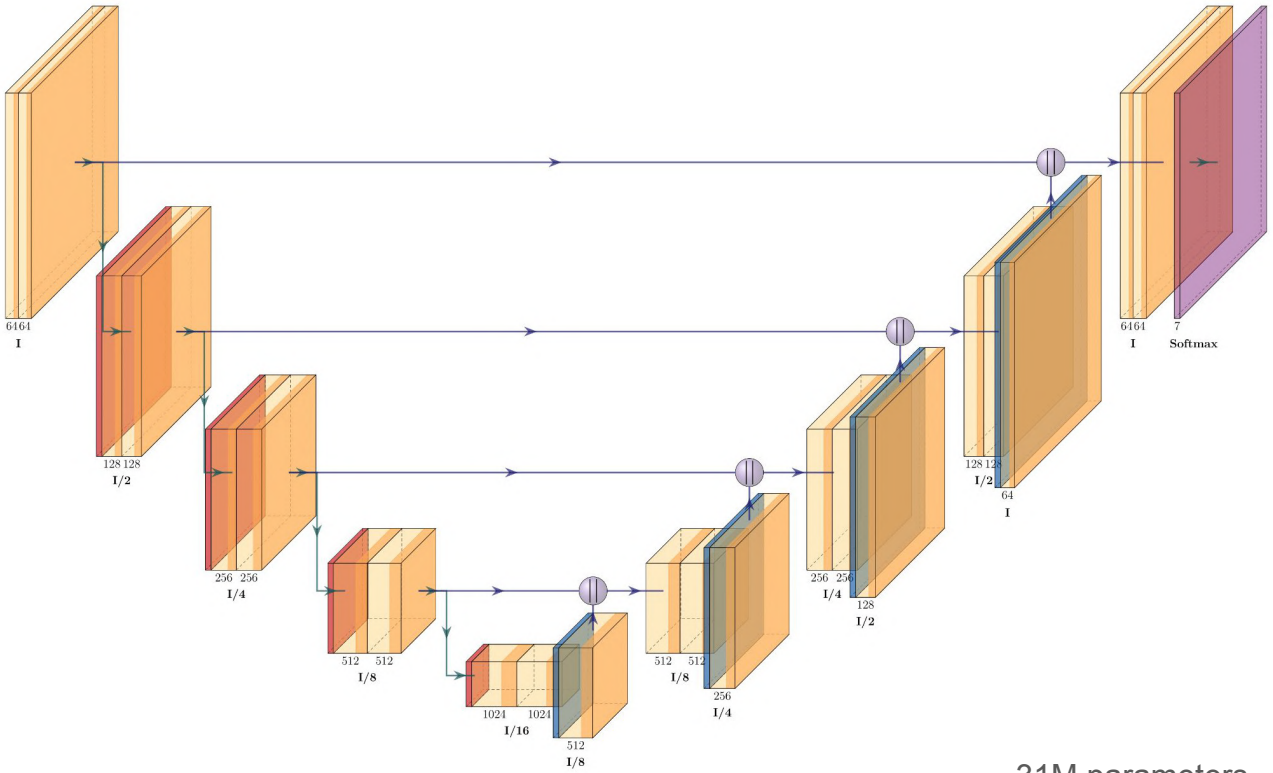
Baseline model: U-Net

Our baseline model is a **U-Net** architecture.

Compared to the original paper, we added:

- batch normalization layers
- zero padding to preserve the input image size

We used a multi-class cross entropy loss.



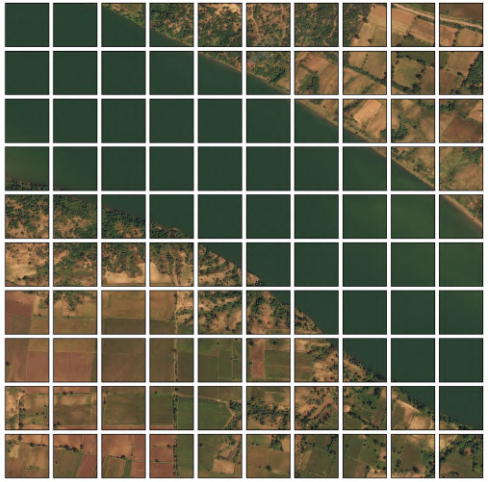
31M parameters

Proposal model: U-Net with patches

The baseline model has a limitation in the batch size that causes a noisy gradient.

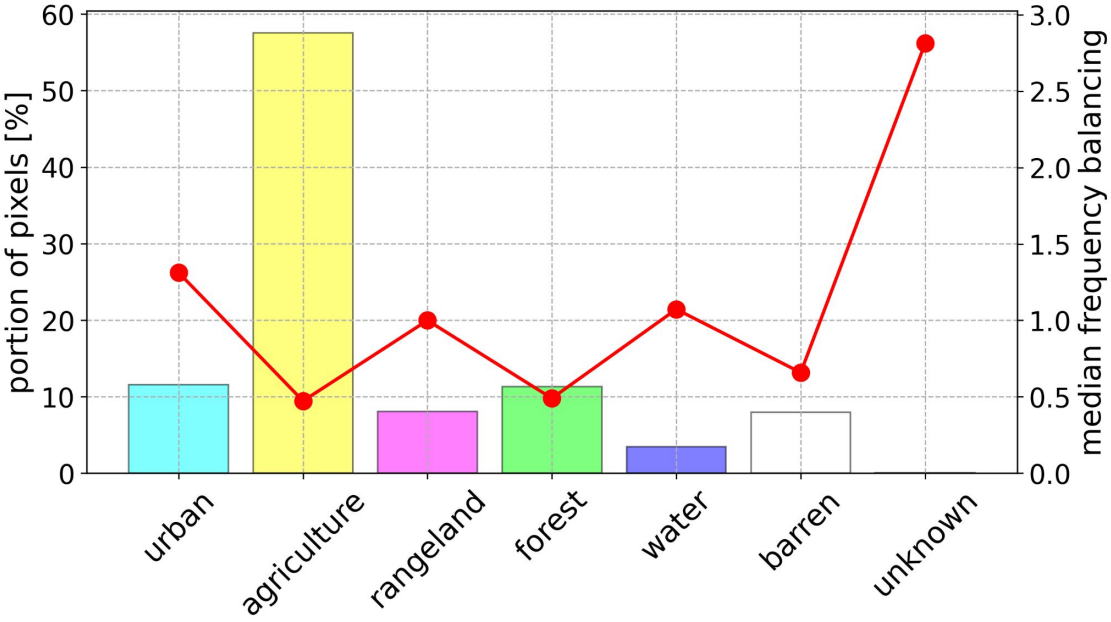
Multi-scale patch-wise pipeline: we rescaled the entire training set using four different scales (0.5, 0.75, 1, 1.25) and divided the images into non-overlapping patches (224x224 pixels).

Set	# patches
Training	60000
Validation	8000
Test	8000



Proposal model: Weighted Cross Entropy

We implemented a multi-class cross entropy loss with different class weights calculated using median frequency balancing.



State-of-the-art model: DeepLabV3+ with patches

DeepLabV3+ has 3 parts:

41M parameters

- encoder
- Atrous Spatial Pyramid Pooling (ASPP) module
- decoder

The encoder is ResNet-101 pre-trained on ImageNet.

The decoder includes bilinear upsampling layers, skip connections and convolutional layers.

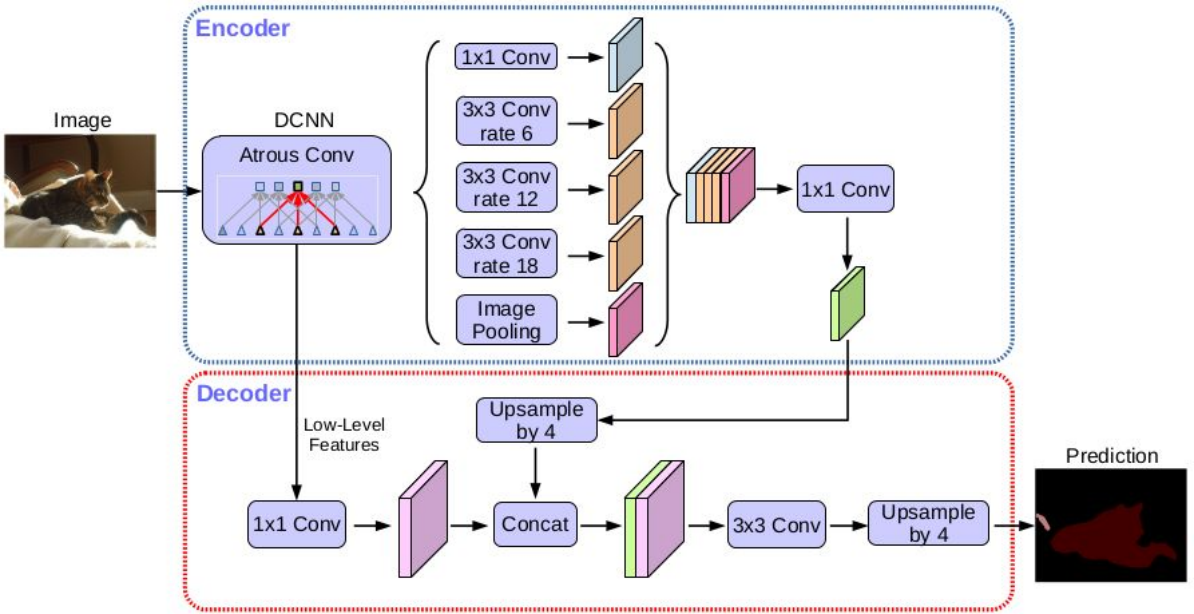


Image taken from DeepLabV3+

State-of-the-art model: ASPP module

The **ASPP module** increases the receptive field applying atrous convolutions with different dilation rates to the feature maps from the encoder.

The outputs are concatenated together also with a global average pooling of the feature maps from the encoder to provide an additional context information.

The decoder and the ASPP module exploit the depthwise separable convolution, resulting in a faster and stronger network.

The loss used is the same weighted multi-class cross entropy as in the proposal model.

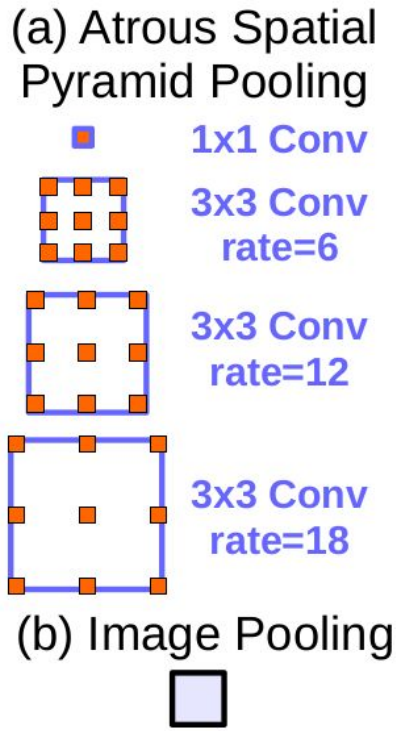


Image taken from DeepLabV3

Training

The optimizer chosen for all the models is Adam with initial learning rate of 10^{-3} and with weight decay parameter of 10^{-5}.

We evaluate the model with this set of metrics:

$$\text{MIoU} = \frac{1}{k-1}\sum_{i=1}^{k-1}\text{IoU}_i$$

$$\text{F1-score} = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

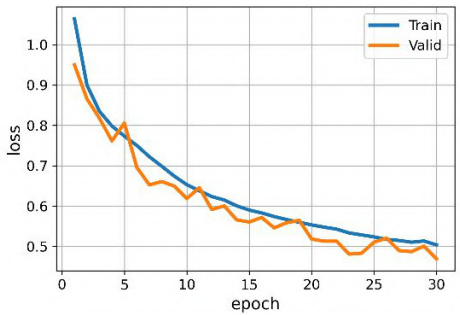
where

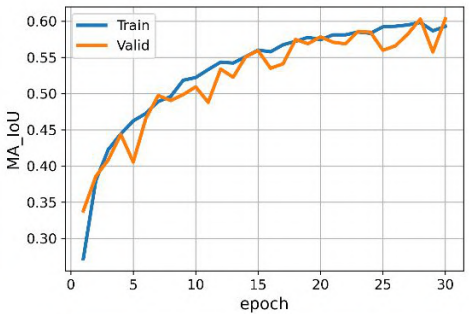
$$\text{IoU}_i = \frac{tp_i}{tp_i + fp_i + fn_i}$$

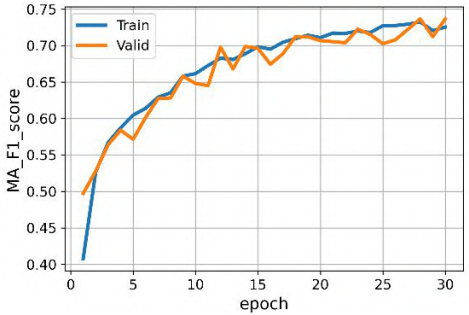
$$\begin{cases} \text{precision} = \dfrac{1}{k-1}\displaystyle\sum_{i=1}^{k-1}\frac{tp_i}{tp_i + fp_i} \\ \text{recall} = \dfrac{1}{k-1}\displaystyle\sum_{i=1}^{k-1}\frac{tp_i}{tp_i + fn_i} \end{cases}$$

Training: Learning curves

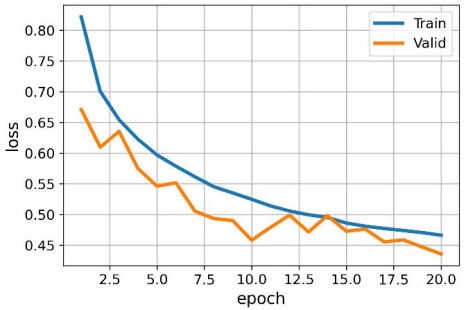
Proposal U-Net

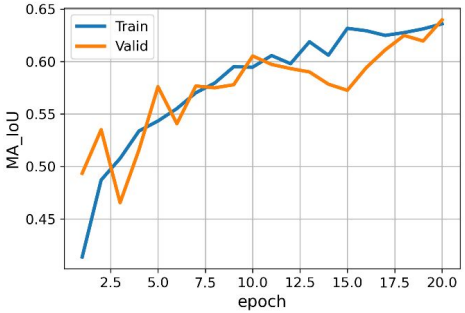


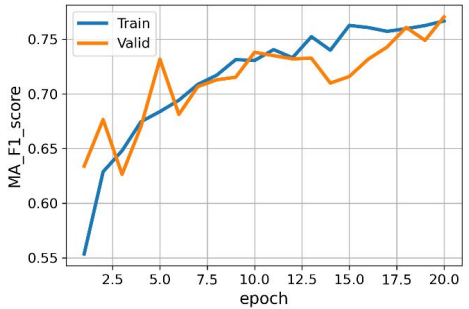




DeepLabV3+

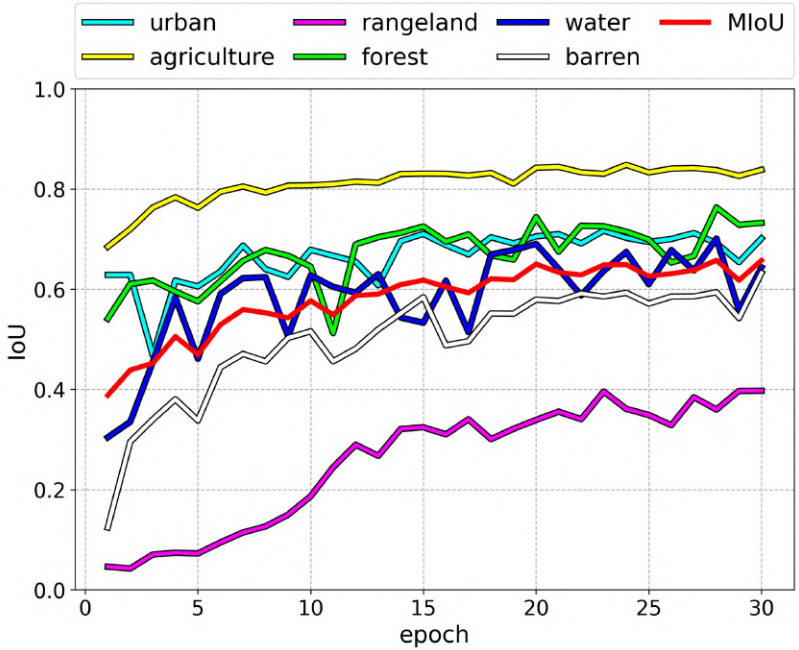




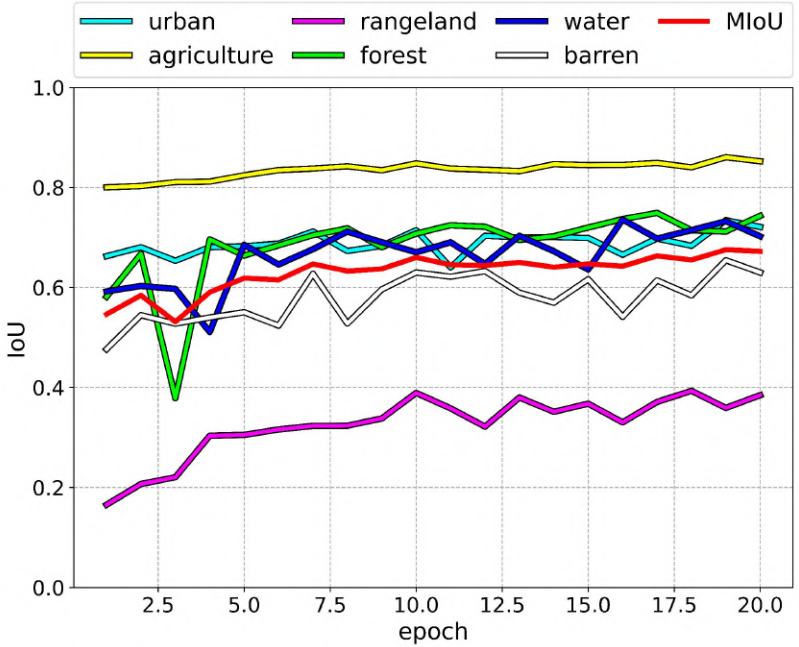


Training: IoU per class

Proposal U-Net



DeepLabV3+



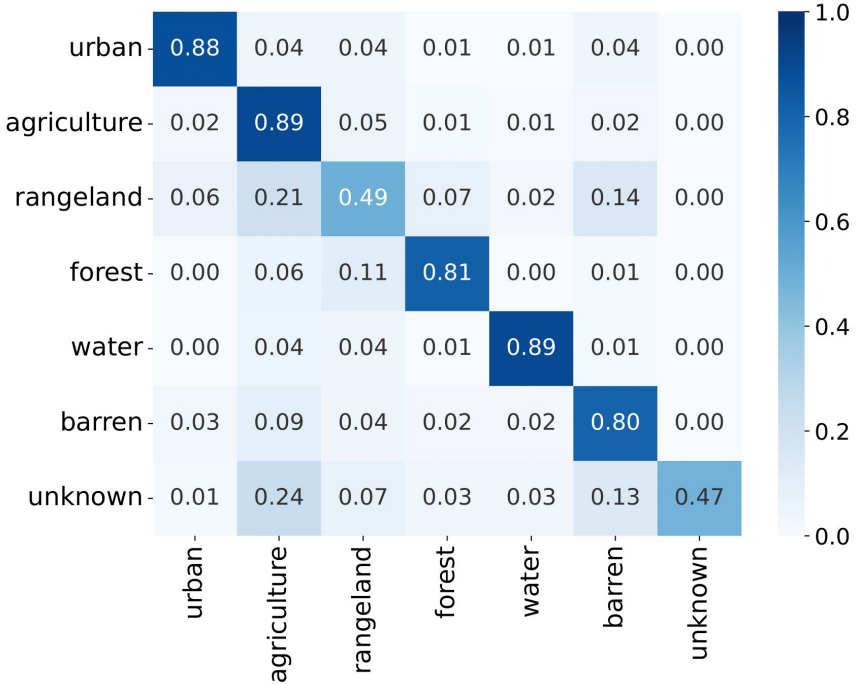
Results: Metrics

Here are the results obtained in the test set:

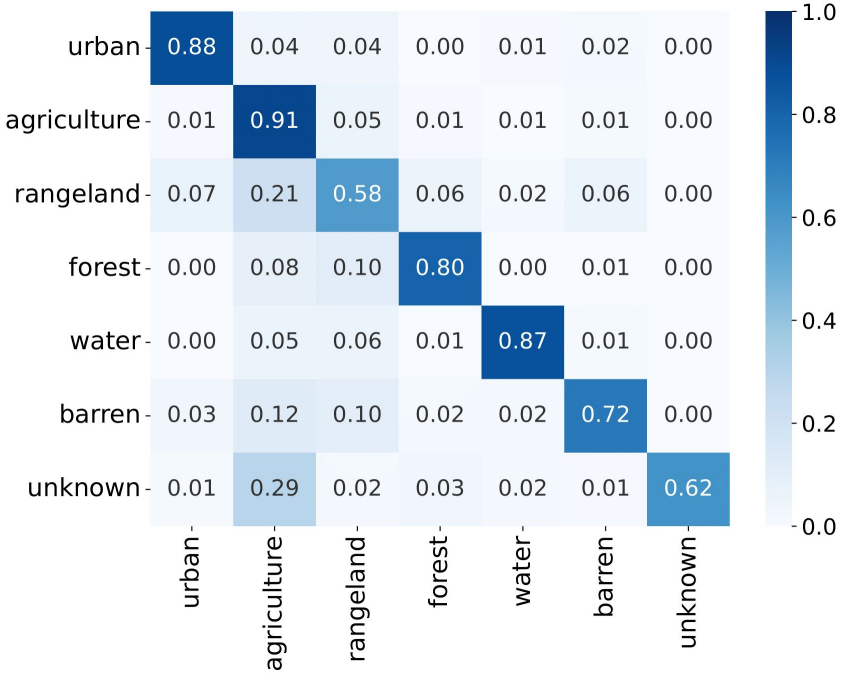
Model	Baseline U-Net	Proposal U-Net	DeepLabV3+
Epochs	175	30	20
Loss	1.05	0.39	0.32
MIoU	0.31	0.65	0.66
F1-score	0.47	0.77	0.78

Results: Confusion matrix

Proposal U-Net



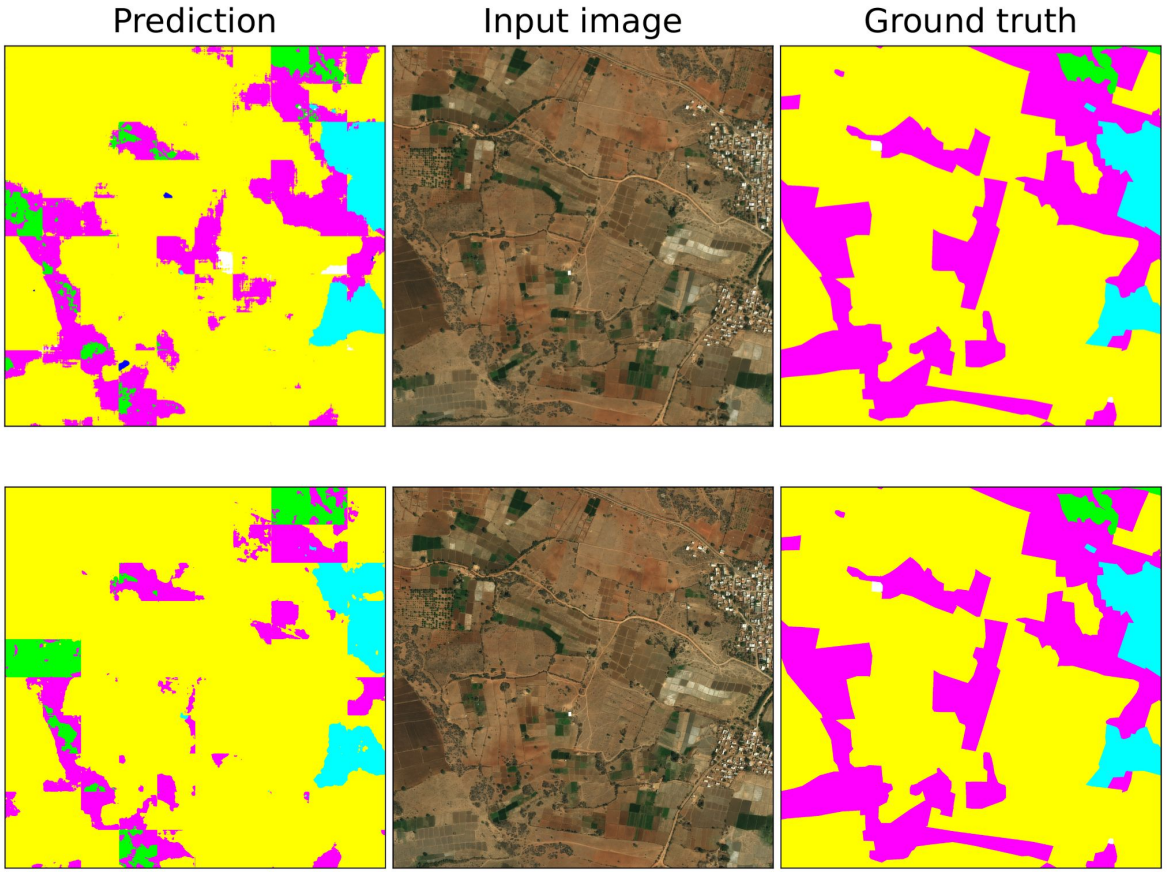
DeepLabV3+



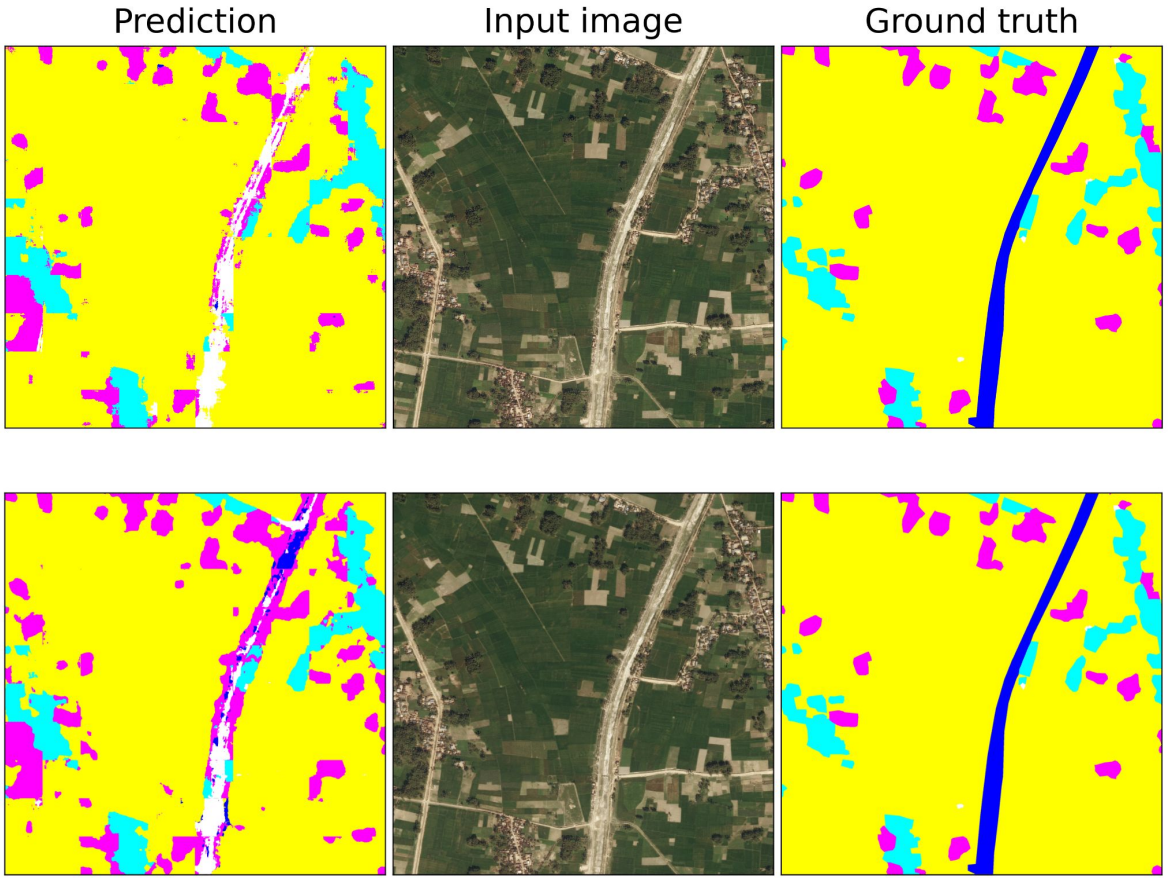
Predictions: Splitting image

Proposal U-Net

DeepLabV3+



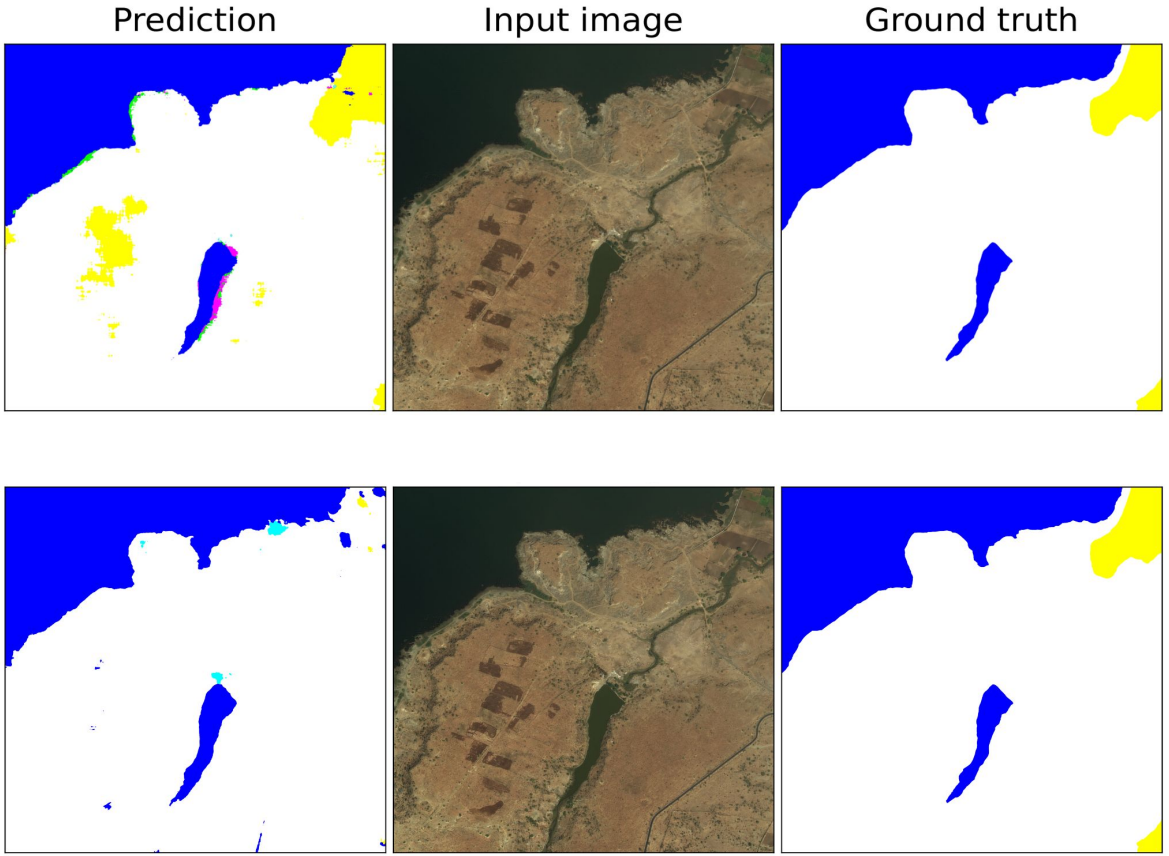
Predictions: Splitting image



Predictions: Full image

Proposal U-Net

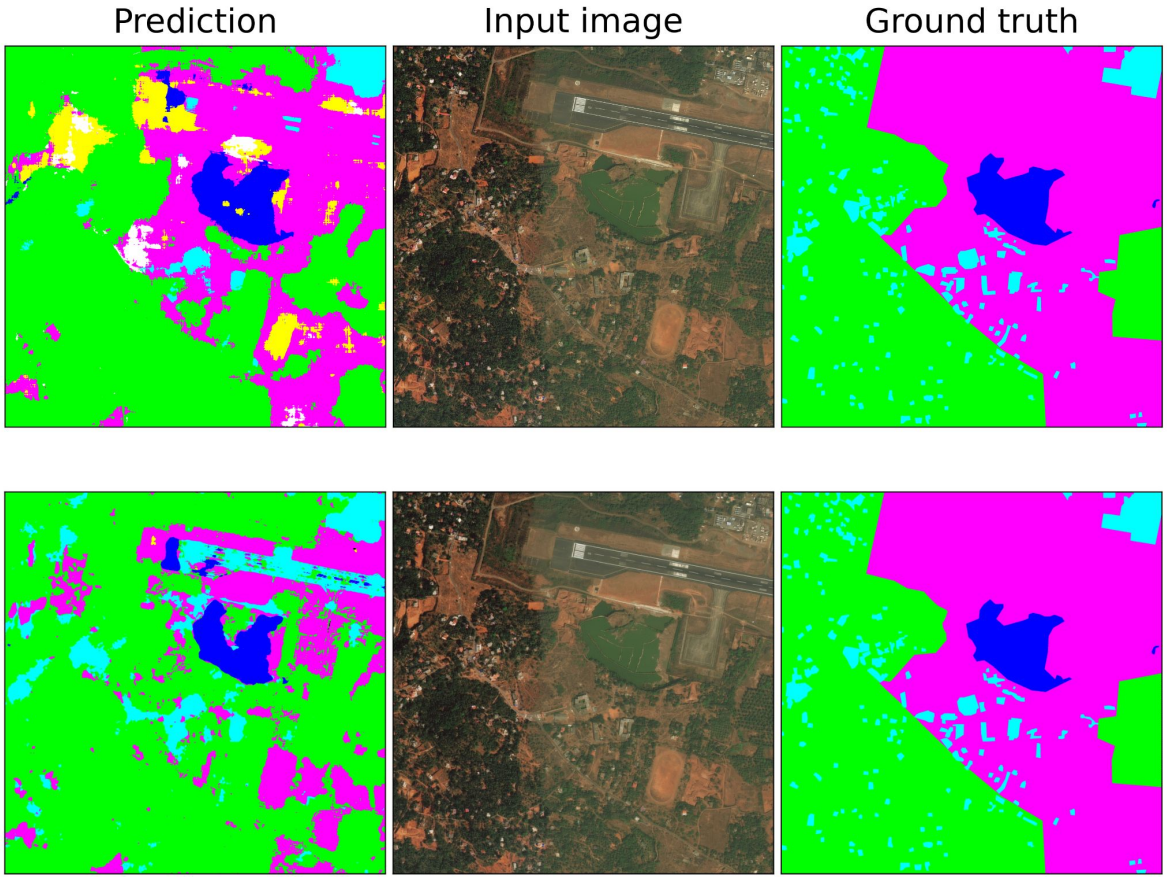
DeepLabV3+



Predictions: Full image

Proposal U-Net

DeepLabV3+

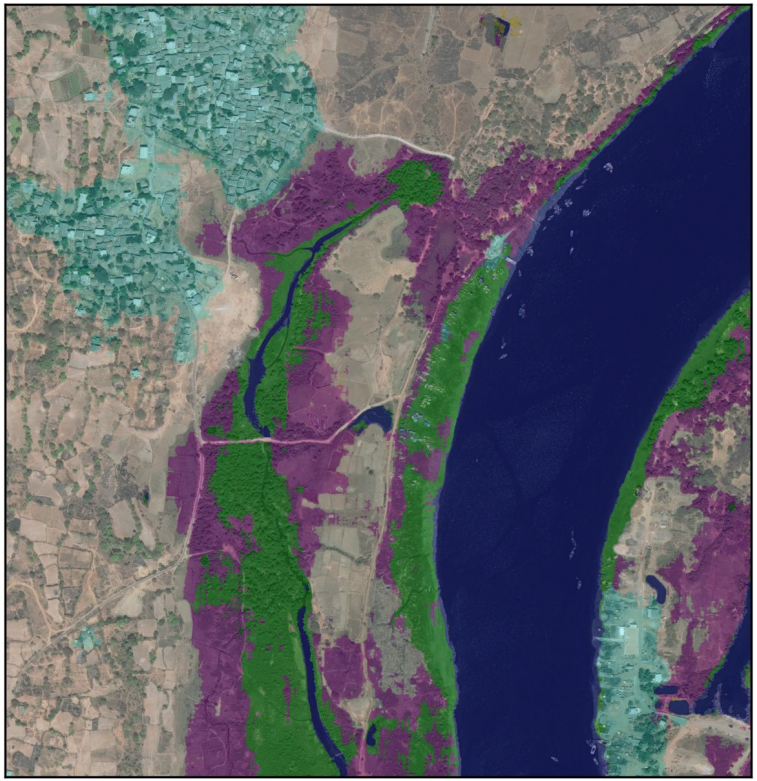


Conclusions

We presented the common challenges faced in remote sensing imagery datasets.

Introducing a multi-scale patch-wise pipeline and applying median frequency balancing to weight the loss function allows our version of U-Net to achieve good performance.

Our results outperform the DeepGlobe challenge baseline score and are competitive with those of the competition winners.



Depthwise Separable Convolution