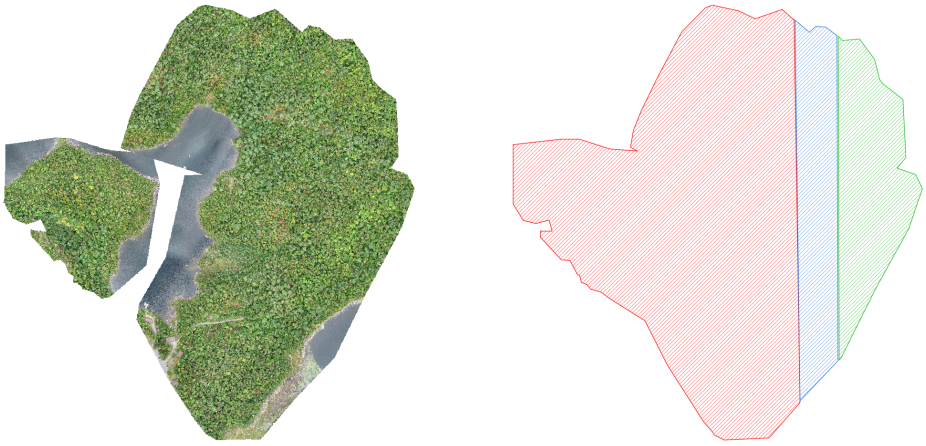
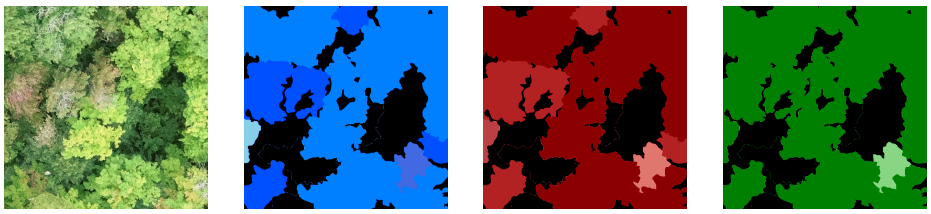


Fig. 7: Spatial splits of the dataset. The image on the left depicts the entire region where the aerial imagery was captured, while the image on the right shows the different subregions used to train, evaluate and test models.The training region is represented by ■, the validation region in ■, and the test region in ■. To prevent data leakage between the subsets, a buffer tile is omitted between the adjacent regions. This spatial partitioning ensures that the model's performance is assessed on geographically distinct areas, simulating real-world scenarios where the model would be applied to unseen locations.



(a) Image **(b)** Species **(c)** Genus **(d)** Higher-level

Fig. 8: Example of the proposed three-level hierarchical label structure. The labels are concatenated to form semantic segmentation masks where each channel correspond to a specific taxonomic level: species 8b, genus 8c and higher-level taxon 8d. In this example, there are three classes at the species and genus level. However, the higher-level taxon only has two classes due to the aggregation of different trees under one class. Note that the colors used in this image do not conform to the color code shown in Table 1.

A Dataset

The Figure 7 shows the training, validation and test split used in our dataset. This has been carefully created to avoid data leakage between the split. Figure 8 shows an example of the hierarchical labels used for training.

B Methods

The single image semantic segmentation experiments are conducted with diverse methods detailed in the following sections.

B.1 Single image semantic segmentation

U-Net U-Net [53] is a widely adopted convolutional neural network (CNN) architecture [19, 20, 38] designed for efficient image segmentation tasks. The architecture consists of an encoder path and a decoder path, which together form a U-shaped structure. The encoder path follows the typical structure of a CNN, consisting of successive CNN layers, rectified linear units (ReLU), and max-pooling operations, which gradually reduce the spatial dimensions while increasing the number of feature maps. The decoder path utilizes transposed convolutions to upsample the feature channels, enabling the network to construct segmentation maps at the original input resolution. The U-Net architecture uses skip connections [27] to concatenate feature maps from the encoder path with the corresponding upsampled feature maps in the decoder path.

DeepLabv3+ The DeepLabv3+ architecture [13] is an image segmentation method built upon strengths of pyramid pooling with an encoder-decoder structure [12]. The encoder module of the DeepLabv3+ utilizes 'dilated' (also named 'atrous') convolutions to extract dense feature maps at multiple scales with larger receptive fields while keeping the computation costs lower. The encoder incorporates atrous spatial pyramid pooling (ASPP), which applies atrous convolutions with different dilation rates in parallel to further capture multi-scale context [12].

The decoder module of the DeepLabv3+ combines the output of the encoder with low-level features from the encoder. This information is refined with 3×3 convolutions to produce the final output segmentation maps.

Mask2Former The Mask2Former architecture [14] is a versatile method that applies binary masks to focus attention only on the areas with foreground features. The architecture consists of three parts: a backbone network, a pixel decoder, and a transformer decoder. Universal backbones (ResNet [27] or Swin Transformer [40]) are used to extract features from the input image. The low-resolution features are then used in a pixel decoder and upsampled to higher resolution. The masked attention is finally applied on the pixel embeddings in the transformer decoder.

To reduce the computational burden of using high-resolution masks, the transformer decoder processes the multi-scale features per resolution one at a time. The Mask2Former architecture performs well across a variety of tasks like semantic, instance, and panoptic segmentation, which makes it a popular choice.

B.2 Time series semantic segmentation

3D-UNet The 3D-UNet method [15] is composed of successive 3D convolutions with a $3 \times 3 \times 3$ kernel, followed by batch normalization and a leaky ReLU activation. The 3D-UNet downsampling part is composed of five blocks, separated by spatial downsampling after the second and fourth blocks. The upsampling part consists of 5 blocks with transposed convolutions while features from the downsampling part are concatenated similarly than U-Net [53].

U-TAE The U-TAE architecture [23] has been introduced for panoptic segmentation of SITS. It consists of three main parts: a multi-scale spatial encoder, a temporal encoder, and a convolutional decoder that produces a single feature map with the same spatial resolution as the input. The sequence of images is processed in parallel by the spatial encoder, and the temporal attention encoder (TAE) is applied at the lowest resolution features to generate attention masks. These masks are interpolated and applied to each feature map, allowing the extraction of spatial and temporal information at multiple scales. The decoder uses a series of transposed convolutions, ReLU, and batch normalization layers to produce the final feature map.

C Experimental Setup

For training our methods, we employ the Adam optimizer [34] for all models except Mask2Former, which is trained with the AdamW optimizer [42] to maintain consistency with the original training methodology. We trained all models with an learning rate of $1e-4$ with exponential learning rate decay for 300 epochs.

We included rotation (in multiples of 90°) with horizontal flips as data augmentation to enhance the diversity of the training data. The batch sizes used for each model are detailed in Table 9. These were set to the largest size that could fit within a NVIDIA RTX 8000 GPU.

We train our models either using our proposed hierarchical loss, noted HLoss, and described in Section 4.3, or using a combination of dice and cross-entropy losses, noted Dice+CE. The latter is a popular choice for segmentation tasks since the dice loss measures the overlap between the predicted and ground truth masks, while the cross-entropy loss quantifies the dissimilarity between the predicted and true class probabilities. We trained the Mask2Former model with the loss function proposed by its authors [14]. This loss function improves

Model	Batch Size
U-TAE	4
Unet-3D	6
Processor+U-Net	16
Processor+DeepLabv3+	16
U-Net	16
DeepLabv3+	16
Mask2former	16

Fig. 9: Batch sizes used for training.

the training efficiency by randomly sampling a fixed number of points in the labels and predictions.

The loss weighing scheme and other implementation details are kept consistent with original implementation to ensure a fair comparison. Note that we did not run Mask2Former with HLoss and Dice+CE loss as the training would be much more computationally expensive, resulting in a smaller batch size.

The performances of our models are evaluated with the Intersection over Union (IoU) metric, also known as the Jaccard index, measuring the overlap between the predicted and ground truth masks. Letting A and B be two sets, the IoU score is defined as:

$$\mathrm{IoU}(A, B) := \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}. \tag{7}$$

The mean IoU (mIoU) is computed by averaging the IoU scores across all classes. This metric provides a comprehensive assessment of the segmentation performances of a model, taking into account both the precision and recall.