Scalable Vision Transformers for Remote Sensing Semantic Segmentation

by

Ezra MacDonald
B.Sc., Vancouver Island University, 2020

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science

in the Department of Computer Science

Scalable Vision Transformers for Remote Sensing Semantic Segmentation

by

Ezra MacDonald
B.Sc., Vancouver Island University, 2020

Supervisory Committee

Dr. Yvonne Coady, **Supervisor**
(Department of Computer Science)

Dr. Sean Chester, **Departmental Member**
(Department of Computer Science)

ABSTRACT

Assessing and monitoring environmental landscapes plays a critical role in preserving the environment and ensuring the well-being of communities around the world. The launch of low-orbit earth observation satellites has dramatically increased the availability and resolution of remote sensing data, enabling more precise and frequent monitoring of environmental changes and human impacts across diverse ecosystems. Traditional manual methods of analyzing this data to measure environmental properties are being improved by deep learning techniques, which can uncover complex patterns within the data. Recently, the Transformer architecture has been extended to computer vision, further enhancing the versatility and scalability of deep learning models.

This thesis investigates the application of the Transformer architecture to semantic segmentation using medium-resolution satellite data. It explores the unique properties of remote sensing data and proposes techniques to improve deep learning model architectures and training methodologies for optimized results. Two contributions are presented: MineSegSAT and VistaFormer.

MineSegSAT is designed to identify and monitor environmentally impacted areas of mineral extraction sites using Sentinel-2 imagery. It incorporates state-of-the-art deep learning models and loss functions to automate the detection of disturbed areas, aiding in environmental compliance monitoring.

VistaFormer is introduced as a lightweight and efficient model for the semantic segmentation of satellite image time series (SITS) data. It features an encoder-decoder architecture with gated convolutions and self-attention Transformers in the encoder, paired with a lightweight convolution decoder. This model is designed to handle noise from atmospheric distortions and cloud cover while maintaining high performance and efficiency.

The experimental results demonstrate that VistaFormer outperforms state-of-the-art models on time series crop-type semantic segmentation benchmarks, using fewer floating point operations and fewer trainable parameters. The findings suggest that Transformer-based architectures can significantly enhance the accuracy and efficiency of satellite imagery analysis, providing valuable tools for environmental and agricultural monitoring.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Semantic segmentation is a foundational computer vision task that assigns a unique class label to each pixel. This technique is essential in remote sensing (RS) for land cover classification, crop type mapping, deforestation detection, disaster management, pollution monitoring, and more. The rapid growth in satellite operations and advancements in deep learning have dramatically increased the use cases and effectiveness of semantic segmentation. Earth observation satellite constellations, such as Sentinel and Landsat, along with high-resolution commercial satellites, have exponentially increased the volume and variety of data accessible to researchers and practitioners. These satellites provide high revisit frequencies and fine spatial resolutions, enabling continuous monitoring and detailed analysis of Earth's surface. This abundance of data, along with advancements in data processing and storage capabilities, allows for more sophisticated and accurate RS semantic segmentation models.

RS semantic segmentation faces several challenges, including dealing with high variability in spatial and spectral resolutions, managing large-scale and heterogeneous datasets, overcoming the effects of atmospheric conditions like clouds and haze, and addressing the complexity of natural landscapes with intricate class boundaries. Addressing these issues presents challenges that span multiple disciplines, requiring advancements in data processing techniques, the development of robust machine learning models, and interdisciplinary collaboration to effectively integrate domain knowledge from environmental science, computer vision, and remote sensing.

This study investigates deep learning applications to Sentinel-2 multi-spectral datasets that include imbalanced classes with excessive background areas and the presence of visual obstructions such as clouds and atmospheric distortions. We investigate existing model architectures and their performance when applied to semantic

segmentation tasks using medium-resolution satellite imagery and propose a new model time-series semantic segmentation architecture designed to filter noise from inputs that contain noisy inputs.

This paper presents two significant contributions to remote sensing and environmental monitoring using advanced deep learning techniques: MineSegSAT [1] and VistaFormer. MineSegSAT is introduced as an architectural framework designed to identify and monitor environmentally impacted areas of mineral extraction sites using Sentinel-2 imagery. This architecture serves as a proof of concept, using state-of-the-art deep learning models and loss functions to automate the detection of changes in disturbed areas, to improve environmental compliance monitoring. Instead of prescribing a singular deep learning model, MineSegSAT explores a range of approaches to enhance spatial understanding and adaptability to different scenarios in mining site monitoring.

VistaFormer presents a simple and flexible architectural framework for the semantic segmentation of satellite image time series (SITS) data. The proposed model uses an encoder-decoder architecture that uses gated convolutions and self-attention Transformers in the encoder paired with a lightweight convolution decoder. Gated convolutions are used to reduce the negative impact of visual obstructions and cloud cover during the downsampling of inputs that often appear in remote sensing data. Inspired by the SegFormer architecture, we use position-free self-attention layers which remove the need to include explicit position codes which can (a) reduce the performance of models when training and testing dataset resolutions differ [2] and (b) make training and applying Transformer-based models require more pre-processing.

The self-attention mechanism traditionally scales quadratically with the input sequence length, which can be computationally expensive for long sequences. To address this, VistaFormer's design allows for the substitution of the standard self-attention layer with more efficient attention mechanisms, such as neighbourhood [3], or window attention [4]. This adaptability enhances the model's efficiency without compromising its performance. In our work, we demonstrate the effectiveness of this approach by successfully swapping the self-attention component with Neighbourhood self-attention, leading to improved model efficiency and scalability. We find that VistaFormer outperforms state-of-the-art models in terms of overall accuracy (oA) and mean Intersection-over-Union (mIoU) while using fewer floating point operations and model parameters. We further demonstrate how this model architecture can be adapted for multiple inputs and demonstrate that the model outperforms the existing

state-of-the-art model for multi-input semantic segmentation in terms of mIoU.

Both MineSegSAT and VistaFormer exemplify innovative approaches in leveraging deep learning to enhance the accuracy and efficiency of satellite imagery analysis, providing valuable tools for environmental and agricultural monitoring.

1.1 Literature Review

The advancement of deep learning techniques has significantly enhanced the capabilities of remote sensing applications, particularly in the areas of environmental monitoring and agricultural analysis. We first introduce deep learning and its applications to computer vision before discussing applications of deep learning to RS and the problems we investigate.

1.1.1 Deep Learning & Computer Vision

The transition from traditional machine learning methods to deep learning in the context of computer vision has been driven by several key factors. Traditional methods, such as support vector machines (SVMs) and random forests, often rely heavily on manual feature extraction and engineering, which can be both time-consuming and prone to human error [5]. These methods struggle to capture complex hierarchical structures present in visual data and lack the complexity to represent sophisticated non-linear relationships [6]. In contrast, deep learning, particularly through the use of convolutional neural networks [7] (CNNs) and more recently, Transformers, allows for end-to-end learning, where the model autonomously learns relevant features directly from raw pixel data. This ability to learn multiple levels of abstraction has led to significant improvements in accuracy and performance. Additionally, advancements in hardware, particularly the development of GPUs and TPUs, have enabled the training of much deeper and more complex models on large datasets, making deep learning approaches more practical and scalable. These advancements have resulted in deep learning models achieving state-of-the-art results across various computer vision tasks, like image classification, object detection, semantic segmentation, and panoptic segmentation; far surpassing the capabilities of traditional machine learning techniques [6].

Attention & Transformers

Sequential machine learning problems, such as language modelling, time series forecasting, sequence prediction, and increasingly, computer vision, necessitate architectures capable of capturing temporal dependencies and tracking context over time. Recurrent Neural Network (RNN) layers, including variants like Gated Recurrent Units (GRUs) [8] and Long Short-Term Memory (LSTM) [9] networks, were intro-

duced to address the limitations of traditional neural networks in handling sequential data and maintaining long-term dependencies. These layers enable the modelling of temporal dynamics by maintaining a hidden state that captures information from previous time steps, making them particularly effective for tasks like natural language processing, time series forecasting, and sequence prediction.

While recurrent layers excel at learning deep representations of sequences, they struggle to process data in parallel and are challenged with learning long-range dependencies. The introduction of attention [10] and the subsequent introduction of the Transformer [11] architecture, improved on this layer by introducing self-attention mechanisms that enable parallel processing of global sequences and capturing long-range dependencies more effectively, enhancing both computational efficiency and the ability to understand complex patterns in data. While self-attention is highly parallelizable, its computational complexity scales quadratically with the size of the input, making encoding images as a raw sequence of pixels prohibitive for most images. ViT [12] introduced the first pure Transformer-based model that achieved state-of-the-art performance in image classification. This model reduced the computational complexity of applying the Transformer to vision by encoding images into patches and treating each patch as a sequence of tokens.

Semantic Segmentation

The goal of semantic segmentation is to segment an input image or sequence of images according to semantic information and predict a class for each pixel from a set of classes. This task plays an important role in a broad range of applications including scene understanding, medical imaging, robot perception, and satellite image segmentation [13]. In deep learning, CNN-based models like FCN [14] and U-Net [15] introduced influential structures that have been influential to many models performing at the state-of-the-art. FCN replaced fully connected linear layers with convolutional layers, enabling end-to-end dense prediction for tasks like semantic segmentation, while U-Net used an encoder-decoder structure and skip-connections, which preserve information and improve the precision of pixel-level predictions.

More recently, self-attention and Transformer architectures have been incorporated into or used purely for semantic segmentation tasks, leveraging their ability to capture long-range dependencies and process global context effectively to enhance segmentation accuracy and performance. PVT [16] introduced a pyramid structure-

based pure self-attention backbone that outperformed comparable CNN-based architectures. PVT was then improved on by models like Swin [17, 4], Twins [18], and CoaT [19] that removed fixed size position embeddings to enhance local feature representations and improve model results on dense prediction. [2] introduced SegFormer, a more efficient alternative, that among other contributions introduced a purely data-driven position encoding layer using 3×3 depth-wise convolutions in the MLP layer of the Transformer. More recent model architectures like Mask2Former [20] and I-JEPA [21] share structural similarities with the original Transformer architecture, by employing self-attention mechanisms to process and compare different parts of the input data.

Video Computer Vision

Unlike static images, video demands the analysis of sequential frames, necessitating the processing of both visual signals and temporal sequences. This introduces new challenges, as multiple frames significantly increase data dimensions, introduce redundancy, and require modelling of motion dynamics. CNN-based models like S3D [22], I3D [23], and Unet3D [24] marked a significant leap in 3D vision with their cutting-edge approaches to spatial-temporal feature extraction, and their methodologies often serve as feature extraction layers for larger more complex models [25]. Model architectures that incorporate recurrent layers or hybrid models like ConvLSTM [26] have also been proposed though they and pure CNN-based architectures are now largely outperformed by Transformer-based models [25]. Transformer-based video model architectures vary largely in terms of how they handle temporal samples. Models like Video-Swin [27] and SwinV2 [4] use hierarchical designs and aggregate spatio-temporal tokens to reduce both the sequence and image length. GroupFormer [28] introduced query-driven compression where a small set of query tokens attends to the entire input sequence to distil essential information from each sample. TimeSformer [29] utilizes local attention by limiting attention to specific neighbourhoods within the video frames, significantly reducing computational demands. MemViT [30] incorporates memory tokens to store and access information from each frame, effectively capturing long-term dependencies in video sequences. While optical RS data often includes cloud obstructions that could result in corrupted temporally downsampled data if not done carefully, using temporal downsampling techniques as have been used in video Transformer-based models can reduce data dimensions and allow

models to capture temporal patterns.

1.1.2 Semantic Segmentation & Environmental Monitoring

Deep learning has broad and significant applications in the context of remote sensing and environmental monitoring while Sentinel-2 data has been useful for land cover and land use mapping, and for improving the automation of environmental monitoring [31]. In the context of monitoring mineral extraction operations, deep learning models and remote sensing data have been used to assess the significance of environmentally impacted areas [32] and to identify unregistered and illegal mining operations [33] [34]. We proceed in this paper by applying two sizes of the SegFormer [2] model architecture, albeit with a Convolution-based encoder for improving model precision, to segment areas of environmentally impacted areas of mineral extraction sites. In the context of remote sensing, the SegFormer model has been used to extract information for water bodies [35], detect buildings using optical remote sensing images [36], segment coastal wetlands from Sentinel-2 data [37], and for performing road segmentation [38].

1.1.3 Satellite Time Series

Advancing to satellite time series data, previous work introduces U-TAE [39] which uses a U-Net architecture with a temporal attention mask that is only computed for the lowest resolution layer and is then upsampled to higher resolution embeddings. These masks are used to collapse the temporal dimension along with a 1D convolution to produce a single map per resolution. We differ from U-TAE [39] most notably by downsampling both spatial and temporal dimensions after the first encoder layer to reduce floating point operations, and by computing spatial attention in each encoder block.

TSViT [40] introduced an architecture inspired by ViT [12], that uses input dates to encode temporal positions and uses separate self-attention Transformer layers for computing attention weights along temporal and spatial dimensions. This architecture is effective but computationally expensive in terms of floating point operations since it does not downsample inputs and computes attention on time and space sequences separately. TSViT [40] also encodes temporal positions using dates, which does not accommodate integrating additional data sources like radar. Most recently, [41] introduced a model architecture that similarly decomposes spatial and temporal

encoding, opting to compute the similarity between a temporal context cluster and temporal input features. The temporal module is used to wrap a 2D segmentation model, allowing for more model flexibility. The pre-trained model trained in their experiments recorded new state-of-the-art mean-Intersection-over-Union (mIoU) results for crop-class segmentation. We do not compare our model's performance to this architecture as we use both mIoU and overall Accuracy (oA) metrics which were not reported on in [41], the smallest model that improves on state-of-the-art mIoU performance is much larger in terms of trainable parameters than our model, and the performance from randomized weights is not recorded in the papers results.

Chapter 2

Problem

In this paper, we explore two distinct semantic segmentation challenges using Sentinel data, both of which involve segmentation problems characterized by significant background regions. The first challenge involves detecting and monitoring regions affected by mining activities using single-image examples. Building on this research, we shift our focus to a more advanced problem: the efficient semantic segmentation of satellite image time series (SITS) data. By utilizing multiple temporal inputs, we develop models that offer more robust and detailed predictions of land characteristics over time. This method addresses the shortcomings of single-image analysis by compensating for variability and obstructions in remote sensing data, leading to more reliable segmentation outcomes [42].

2.1 Monitoring Environmental Impact of Mineral Extraction

The mining industry has seen a considerable expansion in recent years [43] driven by growing demand for raw materials [44] and demand trends indicate this growth will continue [45]. While this sector is important for the industrialization of the global economy, mining sites can have adverse impacts on the immediate and near environment during mining operations and after closure. Identifying environmentally impacted areas of land can benefit regulators and mining operations internally to ensure environmentally conscientious practices are upheld. There is a critical need for an automated, efficient, and accurate system to monitor and assess the environmental impact of mining activities. The primary challenge is to identify and quantify

disturbed areas caused by mining operations using remote sensing data.

2.2 SITS Segmentation

Obtaining multiple samples can enable a model to make predictions based on a more robust set of inputs, though this requires accounting for an additional temporal dimension which increases the dimension of input data and requires altering model architectures to account for the additional dimension. Remote sensing data often includes visual obstructions like cloud coverage and atmospheric distortions like seasonal variation which requires using additional care when considering a sequence of temporal inputs.

An important application of this data is in identifying crop types, which can be used in precision agriculture, estimating agricultural yields, monitoring crop health, understanding food security vulnerabilities, creating climate adaptation strategies, and more. Crops undergo phenological events throughout their growth cycle that can be captured in remote sensing imagery. Capturing a series of these images increases the likelihood that data (a) does not include visual obstructions like cloud coverage, (b) includes more phenological events, and (c) includes unique surface characteristics based on environmental conditions like rain.

To address these challenges, we develop models designed to be more accessible to researchers with limited computational resources that are simpler to train and apply compared with traditional transformer-based approaches as they do not require using any explicit position codes. By leveraging advances in transformer-based architectures, we can create deep learning models that offer superior accuracy for time series data without the prohibitive computational cost often associated with state-of-the-art techniques. By accurately segmenting and analyzing time series data, we can make environmental monitoring more effective, allowing for more informed decisions in areas such as agriculture, climate science, and natural resource management.

Chapter 3

Proposed Solution

3.1 MineSegSAT

To address the need for effective monitoring of environmentally impacted areas from mineral extraction activities, we propose MineSegSAT, a semantic segmentation model trained on Sentinel-2 data that can be used to identify pixels containing environmentally impacted areas of mineral extraction sites. We use the B2, and B3 sized models presented in the original SegFormer [2] paper using a segmentation head with 2D-convolution layers instead of linear layers to improve model precision.

This implementation builds on work introduced in [1] which applies the SegFormer architecture to Sentinel-2 multi-spectral images trained on environmentally impacted areas of mineral extraction sites identified in [46]. We train on a slightly larger dataset than the one used in [1] that includes additional provinces in Canada, though all of the sample regions in this dataset have not been individually scrutinized to verify the type of mining activity that takes place on these sites and whether or not the sites are active as of that date or in the future. Instead, we rely purely on the analysis performed in [46] to correctly identify these mineral extraction regions. This project further investigates a proof of concept for a service that actively monitors areas of mineral extraction sites to detect expansions or contractions of environmentally impacted areas and otherwise assess the impact of these activities.

3.2 VistaFormer

We build on the experimentation done with deep learning models in MineSegSAT and use that as inspiration while designing an encoder-decoder-based semantic segmentation model. The core challenges we address with the proposed model are to (a) make training and applying Transformer-based segmentation models using remote sensing data simpler and (b) introduce mechanisms for filtering noisy inputs as a result of atmospheric distortions and cloud interference to make 3D downsampling of inputs more robust to noise.

The proposed model, VistaFormer, is a self-attention-based model designed to take a careful view from a distance, using a lightweight self-attention encoder-decoder model architecture to output dense predictions. The self-attention layer used in each encoder block in this model uses a depth-wise convolution in each feed-forward layer to include position information for each pixel as used in [2]. Using this layer to encode positions for pixels removes the need to include explicit position codes which can (a) reduce the performance of models when training and testing dataset resolutions differ [2] and (b) make training and applying Transformer-based models require more preprocessing.

For SITS semantic segmentation tasks, we are given an input $\mathbf{X} \in \mathbb{R}^{C \times T \times H \times W}$ and want to output $\mathbf{Y} \in \mathbb{R}^{K \times H \times W}$ where C denotes input channels, H and W correspond to input dimensions, T denotes the number of samples, and K is the number of classes. Observe that for inputs that include only one time-step, we have $T = 1$, which by squeezing the temporal dimension gives us input $\mathbf{X} \in \mathbb{R}^{C \times H \times W}$.

3.2.1 Patch Embedding

Similar to models like Swin [4] and [2] we downsample inputs using convolution layers, though in our case we use 3D-convolutions. To reduce distortions during downsampling from samples that include visual obstructions like cloud coverage and other atmospheric distortions, we use a simple gated (or partial) convolution layer, as shown in Figure 3.3b, in each encoder block. Specifically, we compute a mask for an input x given by $m = \sigma(\phi_m(x))$ where σ denotes the sigmoid function and ϕ denotes a convolution, and multiply the learned mask by the learned feature, $\phi_l(x)$ giving us $y = \phi_l(x)m$. This is similar to the implementation proposed in [47], except we use batch and layer normalization for the first input and rely on only layer normalization used in the Transformer blocks for the following inputs. This convolution mechanism

was introduced for image in-painting, to ensure that masked pixels in an image input are not used to compute convolution outputs [47], which we find similarly suitable for cases where pixels may contain visual obstructions. Given that the applied context for this model is for datasets where individual pixels account for a considerable area, we carefully downsample inputs along spatial dimensions and do not downsample T for the first two layers of the encoder block.

3.2.2 Encoder

Self-Attention

We compute self-attention using tensors with dimensions Q, K, and V, where each tensor has the same dimension of $N \times C$, where $N = H \times W \times S$ and $S = 1$ for single input images, and $S > 1$ for time-series inputs. To compute self-attention we have

$$\text{Attention}(Q, K, V) = \text{Softmax}\frac{QK^T}{\sqrt{d_{head}}})V.$$

The computational complexity of this process is $O(N^2)$, which is prohibitively large for large images and image sequences. To address this bottleneck, the SegFormer model downsamples the sequence length as follows by transforming K and V as follows

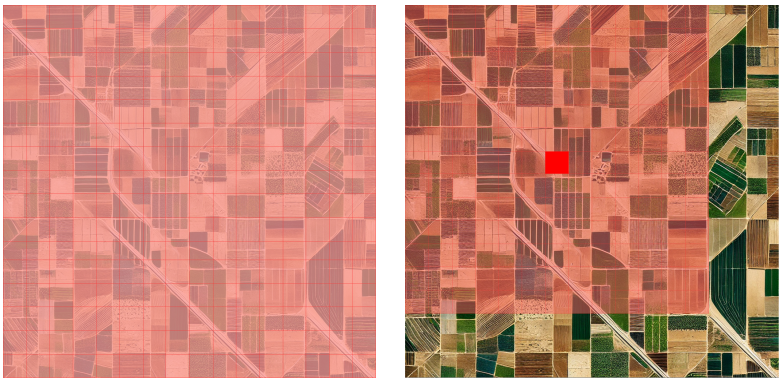
$$\hat{X} = \text{Norm}(\text{Conv}_{R\times R}(X))$$

where X is the sequence to be reduced and Conv uses 2D convolution with patch size and stride of size R to reduce the sequence to dim $\frac{N}{R}$. This output is then projected into linear layers K and V respectively. The new X has dimensions $\frac{N}{R} \times C$ and as a result the complexity of the self-attention mechanism is reduced from $O(N^2)$ to $O(\frac{N^2}{R})$.

In segmentation tasks involving medium to low-resolution satellite data such as Sentinel-2, where the boundaries of interest span areas of 10 meters or less, the use of large sequence reduction ratios or larger convolution patches can significantly impact the model's ability to accurately capture and predict fine details. Each misclassified pixel in these scenarios can have substantial consequences. For this reason, we do not use any sequence reduction techniques in the self-attention layer in the VistaFormer architecture.

To scale VistaFormer's performance, we show that the multi-head self-attention layer used here can be replaced effectively with neighbourhood self-attention [3] which

allows for using a smaller context window than the entire input sequence, which enables computing attention more efficiently.



(a) Global Attention (b) Neighbourhood Attention

Figure 3.1: **(a)** shows an attention layer that uses the entire input sequence in the attention layer which we use as a baseline attention layer, while **(b)** shows neighbourhood attention with a kernel size of 13 that computes attention for each neighbourhood of 13x13 pixels in the input sequence.

Feed-Forward Network

The proposed model does not use absolute or relative positional encoding as in models like ViT or Swin and instead relies on using a 3×3 depth-wise convolution directly in the feed-forward layer to leak location information. This gives us a feed-forward layer defined as follows

$$x_{out} = \text{MLP}(\text{GELU}(\text{Conv}_{3\times3}(\text{MLP}(x_{in})))) + x_{in}$$

where x_{in} is the feature from the self-attention module, and Conv denotes a 3×3 depth-wise convolution which reduces the number of parameters relative to a convolution layer. In the case of other applications of this position-encoding technique, a 2D convolution is used, while for VistaFormer, a 3D convolution is computed against the entire input with dimensions (B, C, T, H, W).

3.2.3 Decoder

The decoder layer shown in Figure 3.3c, upsamples each encoder output using trilinear interpolation to ensure the encoder output has the same dimensions as the input. These outputs are fed through a 3D convolution which extracts the most relevant features from T' and outputs an embedding with dimension (B, C_i, H, W), where C_i is a

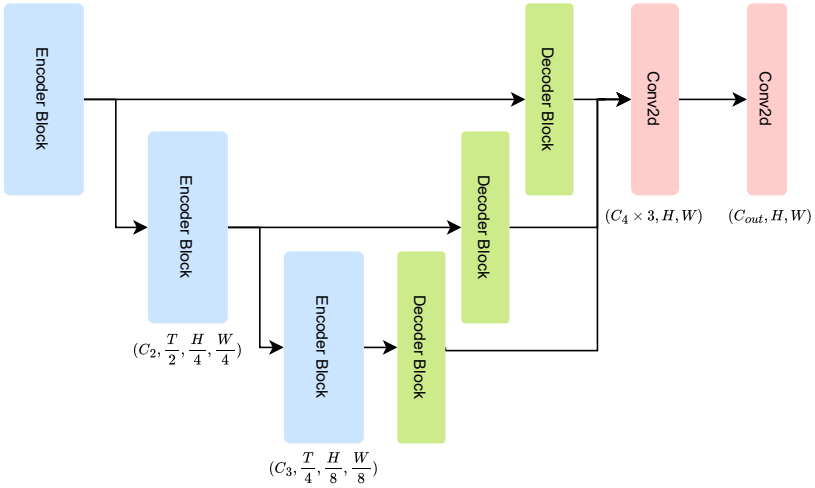
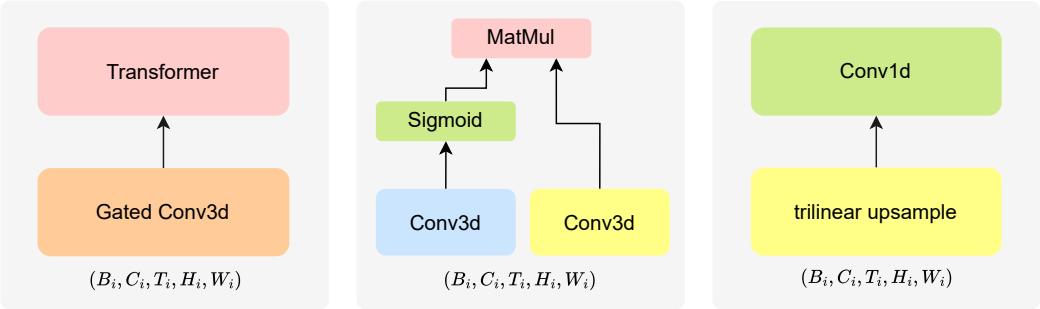


Figure 3.2: The VistaFormer model architecture features a three-layer encoder-decoder architecture. The encoder blocks downsample inputs and processes them with self-attention Transformers, while the decoder blocks upsample the outputs and applies lightweight convolutions to generate dense predictions.



(a) (b) (c)

Figure 3.3: **(a)** VistaFormer's encoder block downsamples inputs using gated convolutions to reduce atmospheric distortions, reshapes them into sequences of tokens, and processes them through self-attention Transformer layers. **(b)** The gated convolution mechanism computes a mask to filter out visual obstructions and multiplies it by the learned feature to reduce input noise. **(c)** The decoder block uses trilinear upsampling and a 1D convolution to extract features and fuse outputs, producing the final dense prediction.

fixed embedding dimension used in all decoder layers. We find using 3D convolutions simpler than using factorized convolutions in its implementation, and in early experiments, more performant. This decoder architecture was designed to strike a balance between simplicity of implementation, using few trainable parameters to increase the likelihood of using this model on smaller GPUs, and carefully upsampling inputs to maximize information retained for the dense prediction. To combine these outputs

we use a 2D convolution to fuse the upsampled encoder outputs before outputting a dense prediction.

Chapter 4

Experiments

For both projects and model architectures we detail the datasets used in experiments, the training parameters and configurations used, and the model architecture and configurations that were used. For MineSegSAT, we additionally detail the data extraction strategy for creating the training and inference datasets. For both projects we further detail some of the unique model and training selections that were adapted to more suitably adapt to the unique segmentation datasets the model is being deployed in.

4.1 Datasets

4.1.1 Mineral Extraction Segmentation

This model was trained using Sentinel-2 tiles, extracted from Amazon Web Services through a collaboration with Earth Daily Analytics which provides atmospherically corrected data on Amazon Web Services. The observation dates for the tiles used for training were from April 1st to September 1st in 2021 for the training period and then a comparison of tiles overlapping the test dataset in 2021 were compared from 2022 for inference and inspection. The tiles that were chosen have less than 1 percent cloud coverage and the tiles within the same coordinate reference system (CRS) were merged using the reverse painter's algorithm. The intersecting ground truth masked data from [46] was re-projected to the CRS of the merged Sentinel-2 tiles and then converted to a raster with 10m resolution to match the resolution of the input tiles.

Given that mining sites in Canada are subject to federal, provincial, and territorial environmental laws and are identified by organizations like Natural Resources Canada

[48], mineral extraction sites across Canada can often be confidently identified in an active monitoring scenario. For this dataset, we include all mining sites identified in [46] in British, Columbia, Alberta, Manitoba, Saskatchewan, Ontario, and Quebec and detail the distribution of tile samples by province in Figure 4.1.

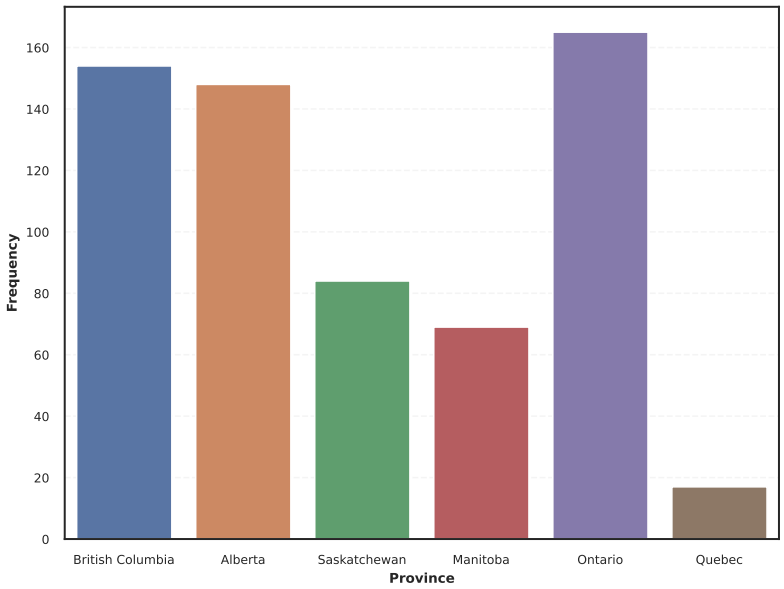


Figure 4.1: Mineral Extraction Site Tile Count by Province. Note that Frequency refers to the number of tiles included in the dataset that are found in the corresponding province.

Each mask and corresponding band file for each period were then split into tiles with dimensions 768x768, 384x384, and 128x128 pixels respective to resolutions of 10m, 20m, and 60m. Note that 8 percent of the pixels included in the dataset used for training, validation, and testing include environmentally impacted areas of mining sites and all 637 of the tiles included in the dataset contain at least 1 pixel that has been environmentally impacted by mining activity.

4.1.2 SITS Segmentation

To evaluate VistaFormer, we use the MTLCC [49] and optical PASTIS [39] semantic segmentation benchmarks. The datasets we chose for evaluating the performance of our model have the following similar noteworthy characteristics (a) they include samples that are obstructed by cloud coverage (in some cases multiple images are entirely covered by clouds), (b) they are both imbalanced datasets with many of

the classes accounting for a very small percentage of the overall pixels, and (c) they include a large number of background pixels that may or may not be easily confused with crop class pixels.

MTLCC

The MTLCC [49] dataset covers an area of 102km × 42km north of Munich, Germany and includes 17 crop classes along with an unknown class that accounts for 39.91% of pixels. The dataset includes 13 Sentinel-2 bands split into 24 × 24 pixels for the highest resolution bands and we up-sample the lower resolution bands using bilinear interpolation to match the dimensions of the highest resolution bands. The dataset includes samples for 2016 which includes 46 samples and 2017 which includes 52 samples. We use the splits provided in the original study for a direct model comparison which has 27k training samples, 8.5k validation samples, and 8.4k test samples. In keeping with the evaluation criteria used in [40], we use 2016 for train, validation, and test datasets. However, we deviate from results reporting from [40, 49], in that we record model results both when the unknown class is included and not included. Not including unknown/background classes during training ensures the model is not penalized for making false predictions in that given area, ensuring the resulting model is more likely to predict false positives, making the model unreliable for predicting realistic boundaries for a class in most applied contexts [50, 51]. Note that the background class in this benchmark accounts for 43.2% of the overall pixels while 13 of the remaining 17 crop classes account for just 13.57% of the overall pixels as shown in Figure 4.2.

PASTIS

The PASTIS [39] dataset spans over 4,000 km^2 with images taken from four regions in France. Each sequence of images includes 10 Sentinel-2 bands split into 128 × 128 pixels and includes between 38 and 61 observations taken between September 2018 and November 2019 [39]. The dataset includes 2,433 samples that are split into 5 folds where for each split, three folds are used for training; one fold is used for validation; and the remaining one fold is used for testing. There are 5 combinations of splits used for measuring model performance on the dataset to ensure that each of the splits can be independently used as the test dataset to better ensure the model generalizes well. The dataset includes 20 classes, with 18 crop types, a background (or non-crop) class,

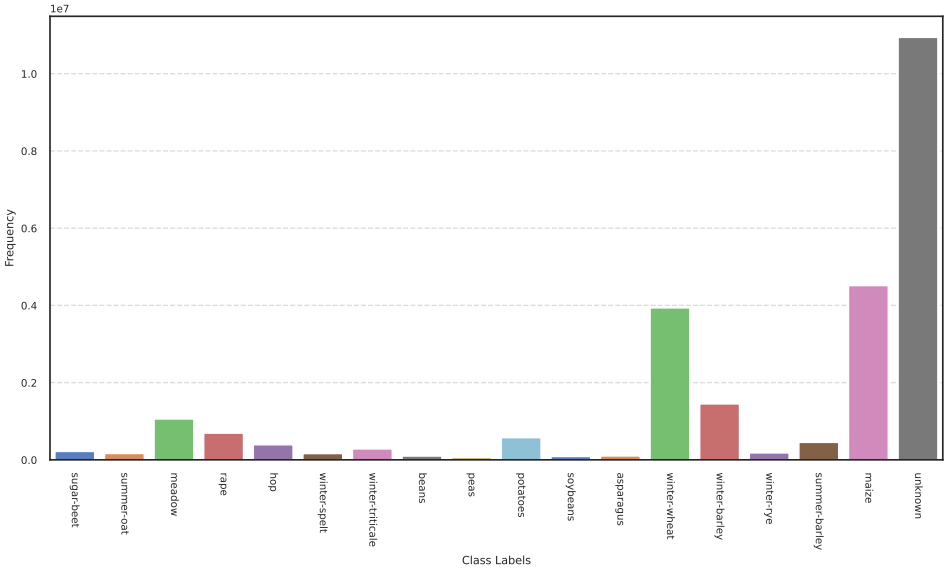


Figure 4.2: MTLCC Class Label Distribution

and a void class which includes either only partial crop class pixels or crop types the authors were unable to confidently identify. The void label is ignored during loss though the background label is included during training and inference results, as specified in [39]. Note that the background class in this benchmark dataset accounts for 39.91% of the overall pixels while 15 of the remaining 18 classes account for 13.86% of the overall pixels as can be seen in Figure 4.3.

4.2 Implementation Details

4.2.1 MineSegSAT

For all models, we train the semantic segmentation models using the weighted Adam optimizer [52] using $\beta_1 = 0.9$ and $\beta_2 = 0.999$ as the coefficients for computing running averages of the gradient and it's square. We use a one-cycle learning rate scheduler that starts with a learning rate of 4×10^{-6} that increases to a max learning rate of 1×10^{-4} and after the first 30% of training and is then reduced to a final learning rate of 1×10^{-6} in the last epoch. From the dataset, we use 70% of samples for training, 15% for validation, and 15% for testing. The models were trained using distributed training on compute nodes with 8 CPUs, 100GB of memory, and two Tesla V100 GPUs for roughly 12-16 hours.

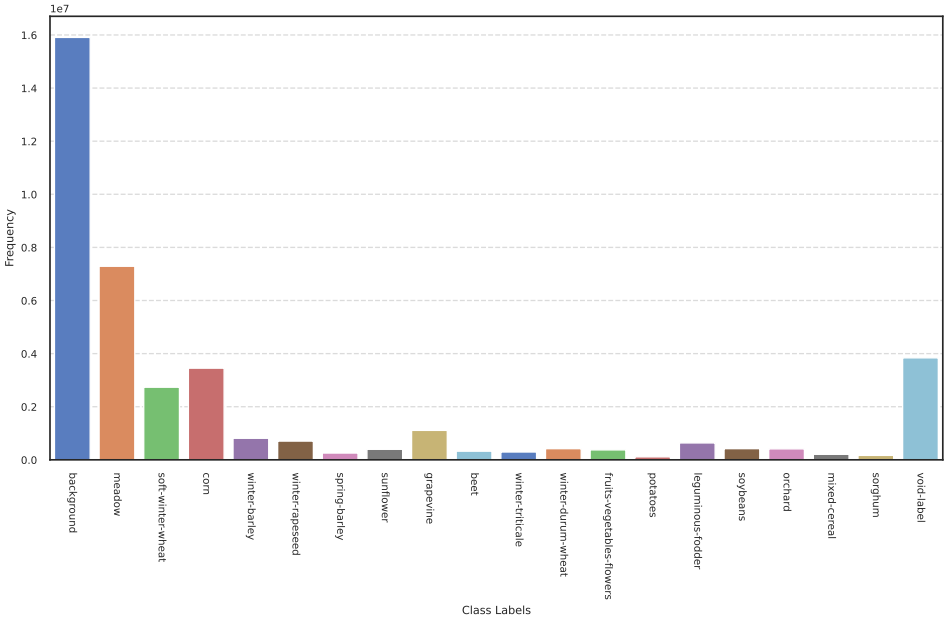


Figure 4.3: PASTIS Class Label Distribution

Train Transforms	Probability
Random Crop	100%
Vertical Flip	50%
Random Rotation	50%
Horizontal Flipping	50%
Channel Shuffle	30%

Figure 4.4: MineSegSAT transformations applied during training

For the SegFormer model architecture, we use a dropout rate of 15% and a drop path of 15%. During training and validation, we use an input image size of 512×512 pixels and a batch size of 4. During training, the transformations provided in Figure 4.4 were applied while only center cropping was applied during validation. While testing the model we split the 768×768 images into patches of 512×512 with an overlap rate of 0.8 and used averaging of the predictions to compute the predict the final segmentation map for the input image. Inputs in all the train, test, and validation stages were scaled using min-max normalization for each band based on metrics computed over the entire dataset. Deviating from, [1], we substitute applying Tversky [53] loss with applying Dice Loss given that the updated dataset is imbalanced, yet still has a considerable representation of the class of interest found in the dataset.

4.2.2 VistaFormer

For both datasets, we train VistaFormer using the weighted Adam optimizer [52] using $\beta_1 = 0.9$ and $\beta_2 = 0.999$ as the coefficients for computing running averages of

the gradient and it's square. We use a one-cycle learning rate scheduler that starts with a learning rate of 0.0004 and increases to a max learning rate of 0.01 after the first 10% of training and is then reduced to a final learning rate of 0.001 in the last epoch. For both datasets, we use a dropout and drop path of 17.5% respectively and use cross-entropy loss for the loss function as in [40, 39]. We found using this higher learning rate and learning rate schedule to outperform lower learning rates for both the max learning rate and the scheduled values. For each input, we normalize using techniques detailed in the original papers [39, 49] and apply flip and 90° rotate transformations with 50% likelihood for each obtained sample input. The models were trained using distributed training on compute nodes with 8 CPUs, 100GB of memory, and two Tesla V100 GPUs for roughly 8-12 hours.

We trained on the PASTIS dataset with a batch size of 32 and a maximum sequence length of 60, and height and width of 32, while for the MTLCC dataset, we used a batch size of 16 and a maximum sequence length of 46 and a provided input height and width of 24. Given that the model uses 3D convolutional layers for upsampling and downsampling which contribute significantly to the number of trainable parameters (relative to our model size); decreasing the input sequence length results in a considerably smaller model. For the MTLCC dataset, we found that decreasing the sequence length from 60 to 46 reduced the number of trainable parameters by 13%. Reducing the sequence dimension for the MTLCC dataset was done in keeping with the sequence length used in [40] and to reduce the number of blank images included with each sample.

Given the small dimensions of the input images for the datasets used during experimentation and the downsampling rate selected for each encoder level, we found that a model architecture with three input layers outperformed other model architectures that included fewer pairs of encoder-decoder blocks. We also found that the selected batch sizes for both the MTLCC and PASTIS benchmarks was optimal relative to smaller or larger batch sizes. The configurations used for the Encoder are given in Table 4.1, while for the decoder, the unique configuration we used for our model was to use 64 output channels for each of the 1D convolution layers which downsample T. We found the attention head dimension outperformed smaller or larger sizes and the selected embedding dimension outperformed larger embedding dimensions at each layer.

For the implementation of VistaFormer that uses 2D Neighbourhood Attention, we use a neighbourhood kernel size of 13 which albeit from a limited ablation analysis

we found to be most performant. We also deviate from the configurations detailed in Table 4.1 by using 1, 2, and 4 attention heads respectively.

Table 4.1: Encoder Configuration

Encoder Layer	Embed Dim	Patch	Stride	Transformer Layers	Attention Heads	MLP Mult
E_1	32	(1, 2, 2)	(1, 2, 2)	2	2	4
E_2	64	(2, 2, 2)	(2, 2, 2)	2	4	4
E_3	128	(2, 2, 2)	(2, 2, 2)	2	8	4

Chapter 5

Evaluation, Analysis and Comparisons

5.1 MineSegSAT

To evaluate the performance of the segmentation models, we use metrics including Precision, Recall, F1-Score, Intersection-over-Union, and Accuracy to better reveal the efficacy of each model's performance. We record results for each of the target classes to further disclose the model's ability to differentiate between the target and background classes. Note that for the below formulas, True Positives are given by TP; False Positives are given by FP; and False Negatives are given by FN.

$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{TP + FN}$$
$$F1_{score} = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

In a comparative analysis of the SegFormer models on the dataset of mineral extraction sites, the smaller B2 model outperformed the larger B3 model. The B2 model, with 49.47 GFLOPs and 24.28 million parameters, achieved higher precision, recall, and F1 scores for the target class and better accuracy and IoU scores for both the target and background classes. Specifically, the B2 model attained a target class F1 score of 59.29 and an IoU of 42.14. In contrast, the B3 model, despite having

Table 5.1: We report semantic segmentation results for input tiles with dimension 768×768 where each input image has been split into patches of 512×512 with an overlap rate of 0.8 and used averaging of the predictions to compute the predict the final segmentation map for the input image. We find that the smaller SegFormer model outperformed the larger model on the dataset of mineral extraction sites.

			Target Class		Background Class	
Model	GFLOPs	Model Params (m)	Precision / Recall / F1	Accuracy / IoU	Precision / Recall / F1	Accuracy / IoU
SegFormer (B2) [2]	49.47	24.28	61.09 / 57.59 / 59.29	57.59 / 42.14	97.65 / 97.96 / 97.81	97.96 / 95.71
SegFormer (B3) [2]	82.90	44.16	54.34 / 57.66 / 55.95	57.66 / 38.84	97.64 / 97.31 / 97.47	97.31 / 95.07

nearly double the computational complexity and model parameters, achieved a lower target class F1 score of 55.95 and an IoU of 38.84. This demonstrates that the smaller model is more effective in this context, possibly due to an improved ability to converge in a shorter training window, given that these models were only trained for 500 epochs down from the 2,000 epochs in [1].

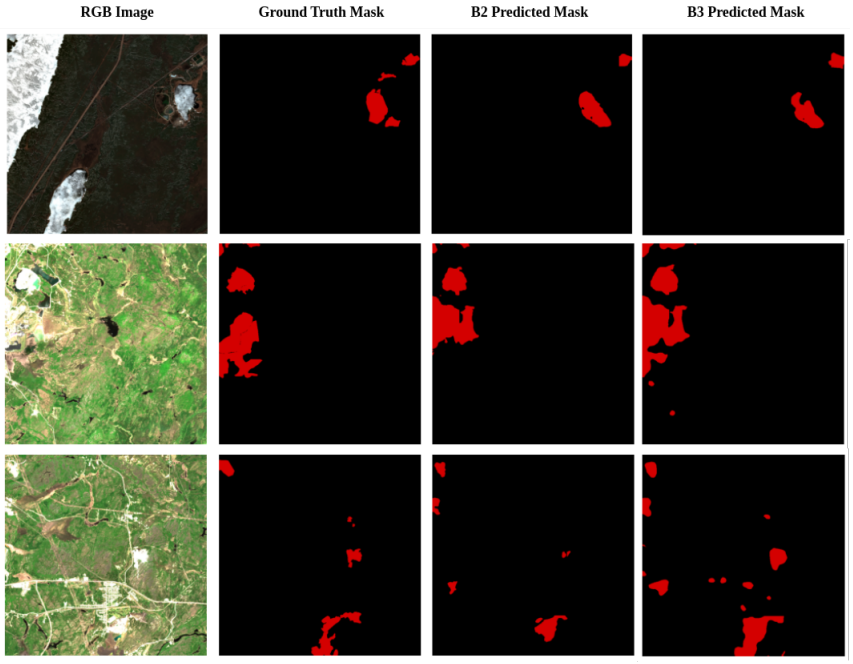


Figure 5.1: We show input and output samples of the B2 and B3 model on the test split of the MineSegSAT dataset. Observe that the B3 model is more likely to predict false positives in the 2nd and 3rd predictions relative to the ground truth which reflects the lesser Precision score of this model relative to the smaller B2 model as found in 5.1.

5.2 VistaFormer

We find that VistaFormer outperforms existing state-of-the-art models in terms of oA and outperforms similarly sized semantic segmentation models in terms of mIoU scores on both the PASTIS and MTLCC on time series crop-type semantic segmentation benchmarks (PASTIS and MTLCC) while using far fewer floating point operations and while using a fraction of the floating point operations than all comparable models and fewer trainable parameters than the current state-of-the-art model. Performance of the model is measured using the mean Intersection over Union (mIoU) score, which computes the averages of the IoU score for each class and the overall Accuracy (oA), which calculates the accuracy summed over all predicted pixels. These metrics were chosen per the metrics used in TSViT [40] and U-TAE [39]. Including oA as well as mIoU in a semantic segmentation task with few classes and many pixels labelled as background or unknown is useful since it provides a straightforward measure of the model's performance across all pixels, helping to ensure that the model effectively distinguishes between relevant and irrelevant regions. For our model, we report the mean and standard deviation performance over three trials for both benchmarks. To estimate GFLOPs for each of the models, we use the FVCore library and estimate the parameter and GFLOPs for each model using an input with shape B=4, T=60, C=10, and H, W=32.

Table 5.2: Comparison with state-of-the-art methods on semantic segmentation. Results for PASTIS were reported for results for fold-1 only, in keeping with [40] and the average of the results for each test set performance across all five folds is given in parenthesis for comparison with [39]. For MTLCC, we report results that exclude the unknown class in training and testing in keeping with [40, 49] and results including the background/unknown class in parenthesis. Note that results marked with an asterisk for PASTIS were trained using the PASTIS dataset with a height and width of 24.

			PASTIS	MTLCC (2016)
Model	GFLOPs	Model Params (m)	oA / mIoU	oA / mIoU
UNET3D [54]	91.10	6.18	82.3 / 60.4*	92.4 / 75.2
UNET3Df [55]	439.78	7.21	82.1 / 60.2*	92.4 / 75.4
U-TAE [39]	23.06	1.09	82.9 / 62.4 (83.2 / 63.1)	93.1 / 77.1
TSViT [40]	91.88	1.67	83.4 / 65.1 (83.4 / 65.4)*	95.0 / 84.8
VistaFormer Neighbourhood (ours)	**4.85**	**1.13**	**83.3 ± 0.1 / 64.8 ± 0.5 (83.7 ± 0.2 / 65.3 ± 0.3)**	**96.1 ± 0.03 / 88.5 ± 0.05 (90.5 ± 0.08 / 79 ± 0.16)**
VistaFormer (ours)	**7.58**	**1.25**	**83.6 ± 0.1 / 64.8 ± 0.2 (84.0 ± 0.1 / 65.5 ± 0.1)**	**95.9 ± 0.14 / 87.8 ± 0.5 (90.4 ± 0.1 / 78.7 ± 0.3)**

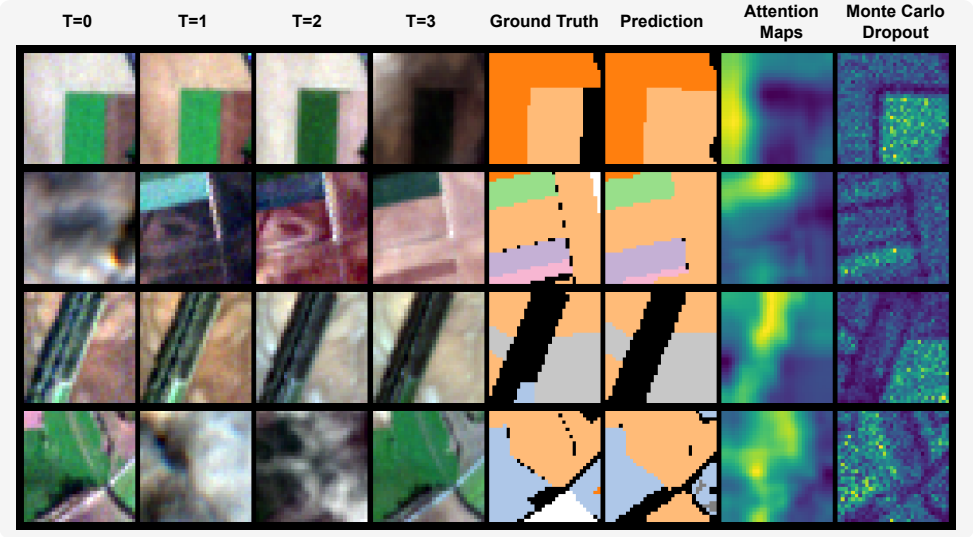


Figure 5.2: VistaFormer sample semantic segmentation predictions on the PASTIS benchmark. Under titles $T = 0, ..., 3$, we show samples of input RGB channels and include these alongside ground truth annotations, model predictions, attention maps, and Monte Carlo dropout [56] predictions to measure the uncertainty of model predictions. We use the dropout settings used during training for Monte Carlo Dropout and the outputs reflect the model certainty measure over 10 iterations. The provided samples highlight that input images are often obstructed by clouds, while the ground truth reflects the imbalance of classes present in the dataset as shown in 4.3.

5.2.1 Ablations

We report ablations concerning (a) decoder layers, (b) encoder layers, and (c) encoder downsampling. We find that using a max pooling layer instead of a 1D convolution in the decoder decreases model performance only slightly across all results excluding overall accuracy for the MTLCC dataset. We find that replacing trilinear upsampling with a transposed 3D convolution improved results inconsistently across datasets while requiring a considerable number of trainable parameters relative to the model size.

For encoder layers, we use gated convolutions by default to reduce input noise and find that not including this layer consistently results in a consistent decrease in both oA and mIoU scores. We introduced a Squeeze and Excitation (SE) [57] layer as an additional convolution filtering mechanism in each of the downsampling encoder layers, though we found inconsistent results in our tests and omitted the layer from the chosen architecture to preserve simplicity. This layer may still be of use for some use cases as it adaptively recalibrates channel-wise feature responses which can reduce

Table 5.3: We present the ablation analysis results for both the PASTIS and MTLCC benchmarks for semantic segmentation. For the MTLCC benchmark, we include the unknown class and for PASTIS we use fold-1 from the PASTIS benchmark which uses folds 1, 2, and 3 for training, fold 4 for validation; and fold 5 for testing. In keeping with Results in Section 5.2, we report the mean and standard deviation for the mIoU and oA scores over three trials for the chosen PASTIS split and the MTLCC dataset.

			PASTIS	MTLCC (2016)
Ablation	Description	Params (millions)	oA / mIoU	oA / mIoU
Encoder Downsampling	$T_1 = \frac{T}{2}$	1.1	83.2 ± 0.01 / 63.4 ± 0.1	90.15 ± 0.01 / 77.4 ± 0.2
	$T_1, T_2, T_3 = T$	1.67	83.5 ± 0.1 / 64.4 ± 0.1	90.4 ± 0.1 / 78.6 ± 0.2
Encoder Layers	w/out Gated Conv	1.16	83.4 ± 0.1 / 64.1 ± 0.2	90.2 ± 0.1 / 78 ± 0.1
	Squeeze & Excitation [57]	1.26	83.5 ± 0.1 / 64.7 ± 0.2	90.3 ± 0.1 / 78.6 ± 0.1
Decoder Layers	Max Pool	0.9	83.3 ± 0.04 / 64.2 ± 0.2	90.3 ± 0.1 / 78.2 ± 0.1
	Conv Transpose	2.37	83.7 ± 0.1 / 64.7 ± 0.1	90.5 ± 0.1 / 78.7 ± 0.2

noise by emphasizing important input features and suppressing irrelevant ones [57].

Concerning encoder downsampling, the base model in Figure 3.2 in effect uses a 2D convolution in the first layer, only downsampling the height and width in keeping with [39]. We find that both downsampling T in the first encoder, giving us $T_1 = \frac{T}{2}$, and not downsampling T in any encoder layer, resulted in decreased model performance relative to our proposed model. These results indicate that downsampling the temporal dimension can be performed effectively for SITS data, contrary to the preservation of the temporal dimension used in [39, 40], allowing our model to reduce the sequence length used in self-attention layers, and subsequently the complexity of the model.

5.2.2 Model Scaleability

For this model architecture, we treat each sequence entry in the temporal dimension T as a unique sample, which increases the floating point operations performed since we compute the attention for each sequence of length $H \times W$, T times. This presents challenges when trying to scale SITS models to larger temporal or input dimensions given that self-attention scales quadratically with the size of the input sequence.

The model used in the core of our experiments uses self-attention with little downsampling, ensuring the model is computationally expensive to scale. To address this, we demonstrate that substituting self-attention with self-attention modules designed to be applied to spatial data improves the model's scalability. More specifically, we find that 2D Neighbourhood attention dramatically reduces the number of floating

point operations required both for small input dimensions and as the spatial dimensions increase as seen in Figure 5.3.

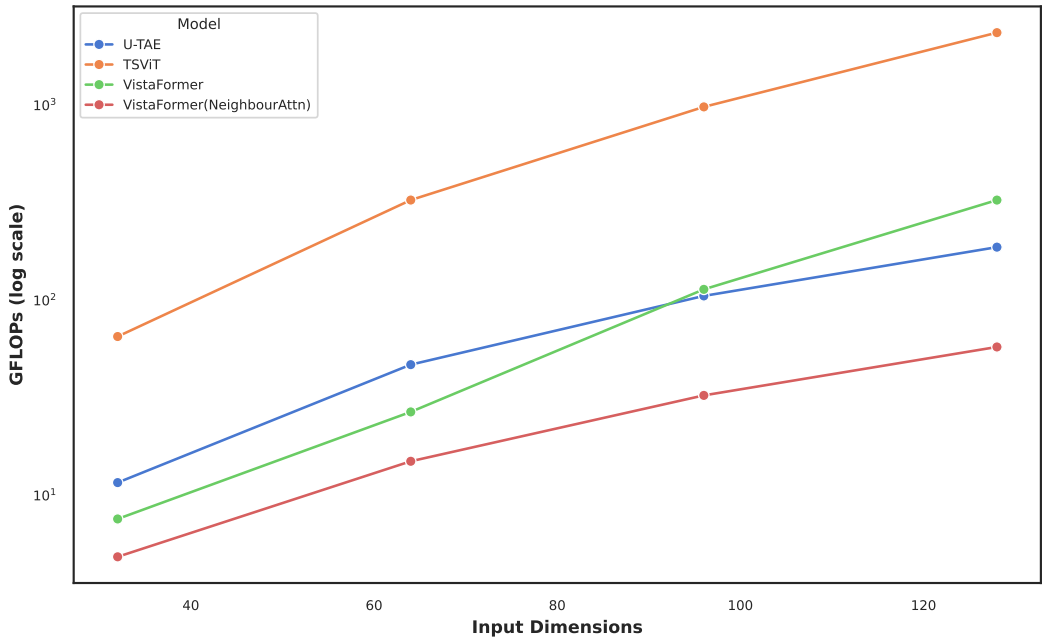


Figure 5.3: VistaFormer with self-attention surpasses the U-TAE model in terms of floating point operations after the input dimension increases beyond a height and width of 64. Note that GFLOPs are scaled logarithmically to more meaningfully show the scalability relationship between TSViT and smaller models. The input dimensions used for computing the GFLOPs are $(B, C, T, H, W) = (4, 10, 30, x_i, x_i)$.

From Figure 5.3, observe that the proposed VistaFormer model architecture faces scaling challenges as the input dimensions or input sequence sizes are increased when using multi-head self-attention. This model requires fewer floating point operations than U-TAE up to input dimensions of size 64 with a fixed temporal dimension of 30 as seen in Figure 5.3, while VistaFormer with 2D-Neighbourhood Attention scales better than all other considered models for larger input dimensions.

5.2.3 Multi-Input SITS

We provide an implementation here that extends the performance of VistaFormer to use multiple inputs. For this model, we use concatenated Sentinel-1 Ascending (S1A) and Sentinel-1 Descending (S1D) data as input for a VistaFormer encoder and use the Sentinel-2 inputs as input for another encoder. The encoder outputs at each layer are concatenated together using a depth-wise convolution layer with an output

Module	**FLOPs**	**Memory**
Self-Attn	$3THWC^2 + 2TH^2W^2C$	$3C^2 + H^2W^2$
Neighbourhood Attn	$3THWC^2 + 2THWCK^2$	$3C^2 + HWK^2$
3D Convolution	$THWC^2K^3$	C^2K^3

Table 5.4: Computation Cost and memory usage of attention patterns and convolutions. This analysis relies heavily on the complexity analysis provided in [3], though we account for the temporal dimension T.

channel dimension of 128 before being fed through the VistaFormer decoder. This architecture is similar to the late-fusion architecture proposed in [58] as each encoder block with respective Sentinel-2 and Sentinel-1 inputs is concatenated channel-wise before being downsampled to a given output dimension.

Table 5.5: Comparison with U-TAE multi-modal model on semantic segmentation with S1 and S2 inputs. We find that our multi-input VistaFormer model outperforms U-TAE in terms of mIoU, while falling slightly behind in terms of oA. Further, the VistaFormer model trained with auxiliary predictors at each encoder layer outperforms VistaFormer with multiple inputs.

		PASTIS
Model	Model Params (m)	oA / mIoU
Multi-Modal U-TAE [58]	1.7	84.2 / 66.3
VistaFormer (multi)	**1.7**	**84.05 ± 0.1 / 66.67 ± 0.15**
VistaFormer (with aux)	**1.7**	**84.08 ± 0.07 / 67.03 ± 0.12**

Chapter 6

Conclusions

For all datasets used in these experiments, we find that the land classes of interest include fine characteristics that would benefit from the inclusion of a richer context. To accomplish this we recommend using satellite data that has a higher resolution than a maximum resolution of 10m per pixel as this would also improve the ability to visually scrutinize model predictions for validation. Further, including additional samples, particularly in the case of the MineSegSAT project, would likely improve the predictive accuracy of semantic segmentation models in all context regions. We proceed by detailing unique findings for each respective project.

6.1 MineSegSAT

The ground truth data and models trained on this data effectively identify the features noted in the original paper. However, further evaluation of the ground truth dataset's quality and enhancements to the model's performance is necessary to ensure its utility for accurate, real-time environmental monitoring. Correlating the identified mining areas with existing publicly identified above-ground or visibly environmentally impacted mining areas could be beneficial for better understanding the efficacy of the data and areas where the model is challenged in making accurate predictions.

Further training could be done that uses a larger dataset and multiple input samples similar to the proposed architecture found in VistaFormer to ensure the model is more robust to input noise. A model trained based on these findings could present value for identifying and assessing multiple risks posed by mineral extraction operations by identifying anomalies in the size variation of tailing ponds, rock

pile formations, open-pit mines, and processing/milling infrastructure which were identified in the ground truth masks. This model would thrive in a circumstance where it is monitoring known active or previously active mineral extraction sites that pose a potential threat to the environment. Given that this data is made available by government agencies [48], implementing a meaningful deep learning-based monitoring service of this nature is not only possible but could significantly benefit the environmental monitoring of mineral extraction sites.

6.2 VistaFormer

We have demonstrated a lightweight SITS semantic segmentation model that achieves efficiency by (a) downsampling both spatial and temporal dimensions early, (b) employing gated convolutions to filter out noise, (c) using position-free self-attention layers to simplify the architecture, and (d) incorporating trilinear upsampling in the decoder to reduce computational complexity. Further, the position-free self-attention layers make this model extensible and simpler to use than existing models. We find that these techniques along with a carefully selected model architecture outperform state-of-the-art models consistently on semantic segmentation benchmarks in terms of oA and mIoU performance while using a fraction of the number of floating-point operations and fewer trainable parameters than other current approaches. The ablation analysis highlights the importance of carefully selecting downsampling strategies and maintaining simplicity in the model architecture to achieve optimal performance.

Some of the approaches in this model are generalizable. We find that gated convolutions are crucial for enhancing performance by filtering noise from input data, and recommend further exploration of these noise reduction techniques in other remote sensing models. Similarly, our proposed architecture introduces lightweight design patterns that can be adapted for different image time series segmentation tasks. The efficiency of this model also makes it suitable for deployment in embedded systems, enabling real-time processing and analysis in resource-constrained environments.

Building on the strengths of our current SITS semantic segmentation model, several follow-up research tasks are proposed to expand its capabilities and refine its performance:

1. **Expanding to Panoptic Segmentation** - To further explore the model's versatility, we propose applying the architecture to panoptic segmentation tasks.

This would allow for assessing the model's ability to perform more precise segmentation tasks by identifying both object categories and individual object instances within a scene.

2. **Adaptation to Higher Resolution Tasks** - Given the model's success in efficiently handling medium-resolution satellite data; adapting and applying the model to higher-resolution inputs is a logical next step. This requires fine-tuning the model's architecture to maintain performance and efficiency while managing larger more detailed datasets.

3. **Experimentation with Additional Attention Mechanisms**: The current use of position-free self-attention layers offers a strong foundation. To enhance the model's capability, experimenting with different types of attention mechanisms such as deformable attention, window attention, and various configurations for neighbourhood attention is recommended. These experiments would explore how different attention mechanisms impact the model's performance in handling complex spatial relationships and dynamic temporal changes.

4. **Noise Reduction Techniques**: While gated convolutions have been effective at noise filtering, further investigation into noise reduction techniques could be performed to compare and contrast or introduce new noise reduction techniques that might reduce input noise. Specifically, developing methods that minimize the impact of distortions in down-sampled data can significantly improve the quality of the model's output, particularly in scenarios where data quality is compromised.

These proposed tasks could enhance the model's current capabilities and extend the applicability of the proposed architecture to more complex and demanding real-world applications. The ultimate goal is to refine the model's architecture to handle a broader range of segmentation challenges, thereby solidifying its utility in practical and meaningful environments.

Bibliography

[1] Ezra MacDonald, Derek Jacoby, and Yvonne Coady. Minesegsat: An automated system to evaluate mining disturbed area extents from sentinel-2 imagery. *arXiv preprint arXiv:2311.01676*, 2023.

[2] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. In *Advances in Neural Information Processing Systems*, November 2021.

[3] Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood Attention Transformer. In *2023 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6185–6194, 2023.

[4] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin Transformer V2: Scaling Up Capacity and Resolution. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12009–12019, 2022.

[5] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015.

[6] Laith Alzubaidi, Jinglan Zhang, Amjad J. Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, J. Santamaría, Mohammed A. Fadhel, Muthana Al-Amidie, and Laith Farhan. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1):53, March 2021.

[7] Y. Le Cun, B. Boser, J. S. Denker, R. E. Howard, W. Habbard, L. D. Jackel, and D. Henderson. Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems 2*, pages 396–404. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, June 1990.

[8] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics.

[9] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, November 1997.

[10] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv preprint arXiv:1409.0473*, September 2014.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 6000–6010, Red Hook, NY, USA, December 2017.

[12] A. Dosovitskiy, L. Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, G. Heigold, S. Gelly, Jakob Uszkoreit, and N. Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv preprint arXiv:2010.11929*, October 2020.

[13] Yujian Mo, Yan Wu, Xinneng Yang, Feilin Liu, and Yujun Liao. Review the state-of-the-art technologies of semantic segmentation based on deep learning. *Neurocomputing*, 493:626–646, July 2022.

[14] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. *arXiv preprint arXiv:1411.4038*, March 2015.

[15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv preprint arXiv:1505.04597*, May 2015.

[16] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 548–558. IEEE, October 2021.

[17] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, October 2021.

[18] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the Design of Spatial Attention in Vision Transformers. In *Advances in Neural Information Processing Systems*, volume 34, pages 9355–9366, 2021.

[19] Weijian Xu, Yifan Xu, Tyler Chang, and Zhuowen Tu. Co-Scale Conv-Attentional Image Transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9961–9970. IEEE, October 2021.

[20] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention Mask Transformer for Universal Image Segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1280–1289, June 2022. ISSN: 2575-7075.

[21] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture. *arXiv preprint arXiv:2301.08243*, April 2023.

[22] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking Spatiotemporal Feature Learning: Speed-Accuracy Trade-offs in Video Classification. *arXiv preprint arXiv:1712.04851*, July 2018.

[23] João Carreira and Andrew Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733, July 2017. ISSN: 1063-6919.

[24] Özgün Çiçek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. *arXiv preprint arXiv:1606.06650*, June 2016.

[25] Javier Selva, Anders S. Johansen, Sergio Escalera, Kamal Nasrollahi, Thomas B. Moeslund, and Albert Clapés. Video Transformers: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):12922–12943, November 2023.

[26] Xingjian SHI, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun WOO. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

[27] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video Swin Transformer. *arXiv preprint arXiv:2106.13230v1*, June 2021.

[28] Shuaicheng Li, Qianggang Cao, Lingbo Liu, Kunlin Yang, Shinan Liu, Jun Hou, and Shuai Yi. GroupFormer: Group Activity Recognition with Clustered Spatial-Temporal Transformer. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13648–13657, October 2021.

[29] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is Space-Time Attention All You Need for Video Understanding? *arXiv preprint arXiv:2102.05095v4*, February 2021.

[30] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale Vision Transformers. *arXiv preprint arXiv:2104.11227*, April 2021. arXiv:2104.11227.

[31] Darius Phiri, Matamyo Simwanda, Serajis Salekin, Vincent R. Nyirenda, Yuji Murayama, and Manjula Ranagalage. Sentinel-2 Data for Land Cover/Use Mapping: A Review. *Remote Sensing*, 12(14):2291, January 2020.

[32] Saki Gerassis, Eduardo Giráldez, María Pazo-Rodríguez, Ángeles Saavedra, and Javier Taboada. AI Approaches to Environmental Impact Assessments (EIAs) in the Mining and Metals Sector Using AutoML and Bayesian Modeling. *Applied Sciences*, 11(17):7914, January 2021.

[33] Remis Balaniuk, Olga Isupova, and Steven Reece. Mining and Tailings Dam Detection in Satellite Imagery Using Deep Learning. *Sensors (Basel)*, 20(23):6936, December 2020.

[34] Aneesh Rangnekar and Matthew Hoffman. Learning representations to predict landslide occurrences and detect illegal mining across multiple domains. In *Climate Change AI*. Climate Change AI, June 2019.

[35] Xiao Yang, Mingwei Chen, Chengjun Yu, Haozhe Huang, Xiaobin Yue, Bei Zhou, and Ming Ni. WaterSegformer: A lightweight model for water body information extraction from remote sensing images. *IET Image Processing*, 17(3):862–871, February 2023.

[36] Meilin Li, Jie Rui, Songkun Yang, Zhi Liu, Liqiu Ren, Li Ma, Qing Li, Xu Su, and Xibing Zuo. Method of Building Detection in Optical Remote Sensing Images Based on SegFormer. *Sensors*, 23(3):1258, January 2023.

[37] Xufeng Lin, Youwei Cheng, Gong Chen, Wenjing Chen, Rong Chen, Demin Gao, Yinlong Zhang, and Yongbo Wu. Semantic Segmentation of China's Coastal Wetlands Based on Sentinel-2 and Segformer. *Remote Sensing*, 15(15):3714, January 2023.

[38] Tian Ma, Xinlei Zhou, Runtao Xi, Jiayi Yang, Jiehui Zhang, and Fanhui Li. A Semi-supervised Road Segmentation Method for Remote Sensing Image Based on SegFormer. In Shuo Yang and Huimin Lu, editors, *Artificial Intelligence and Robotics*, Communications in Computer and Information Science, pages 189–201, Singapore, 2022. Springer Nature.

[39] Vivien Sainte Fare Garnot and Loic Landrieu. Panoptic Segmentation of Satellite Image Time Series With Convolutional Temporal Attention Networks. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4872–4881, 2021.

[40] Michail Tarasiou, Erik Chavez, and Stefanos Zafeiriou. ViTs for SITS: Vision Transformers for Satellite Image Time Series. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10418–10428, 2023.

[41] Xin Cai, Yaxin Bi, Peter Nicholl, and Roy Sterritt. Revisiting the Encoding of Satellite Image Time Series. *arXiv preprint arXiv:2305.02086*, September 2023. arXiv:2305.02086 [cs].

[42] Cristina Gómez, Joanne C. White, and Michael A. Wulder. Optical remotely sensed time series data for land cover classification: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 116:55–72, June 2016.

[43] Victor Maus, Stefan Giljum, Dieison M. da Silva, Jakob Gutschlhofer, Robson P. da Rosa, Sebastian Luckeneder, Sidnei L. B. Gass, Mirko Lieber, and Ian McCallum. An update on global mining land use. *Sci Data*, 9(1):433, July 2022.

[44] Manfred Lenzen, Arne Geschke, James West, Jacob Fry, Arunima Malik, Stefan Giljum, Llorenç Milà i Canals, Pablo Piñero, Stephan Lutter, Thomas Wiedmann, Mengyu Li, Maartje Sevenster, Janez Potočnik, Izabella Teixeira, Merlyn Van Voore, Keisuke Nansai, and Heinz Schandl. Implementing the material footprint to measure progress towards Sustainable Development Goals 8 and 12. *Nat Sustain*, 5(2):157–166, February 2022.

[45] UN IRP. Global resources outlook 2019: Natural resources for the future we want. Technical report, United Nations Environment Programme, 2019.

[46] Liang Tang and Tim T. Werner. Global mining footprint mapped from high-resolution satellite imagery. *Commun Earth Environ*, 4(1):1–12, April 2023.

[47] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas Huang. Free-Form Image Inpainting with Gated Convolution. *arXiv preprint arXiv:1806.03589*, October 2019. arXiv:1806.03589 [cs].

[48] Natural Resources Canada Government of Canada. Natural Resources Canada. The Atlas of Canada. Minerals and Mining in Canada, September 2021.

[49] Marc Rußwurm and Marco Körner. Multi-Temporal Land Cover Classification with Sequential Recurrent Encoders. *ISPRS International Journal of Geo-Information*, 7(4):129, April 2018.

[50] Fengyu Yang and Chenyang Ma. Sparse and Complete Latent Organization for Geospatial Semantic Segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1799–1808, June 2022. ISSN: 2575-7075.

[51] Zhuo Zheng, Yanfei Zhong, Junjue Wang, and Ailong Ma. Foreground-Aware Relation Network for Geospatial Object Segmentation in High Spatial Resolution Remote Sensing Imagery. *arXiv preprint arXiv:2011.09766v1*, November 2020.

[52] I. Loshchilov and F. Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, November 2017.

[53] Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, and Ali Gholipour. Tversky Loss Function for Image Segmentation Using 3D Fully Convolutional Deep Networks. In Qian Wang, Yinghuan Shi, Heung-Il Suk, and Kenji Suzuki, editors, *Machine Learning in Medical Imaging*, Lecture Notes in Computer Science, pages 379–387, Cham, 2017. Springer International Publishing.

[54] Rose M Rustowicz, Robin Cheong, Lijing Wang, Stefano Ermon, Marshall Burke, and David Lobell. Semantic Segmentation of Crop Type in Africa: A Novel Dataset and Analysis of Deep Learning Methods. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 75–82, 2019.

[55] Michail Tarasiou, Riza Alp Güler, and Stefanos Zafeiriou. Context-Self Contrastive Pretraining for Crop Type Semantic Segmentation. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–17, 2022.

[56] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1050–1059. PMLR, June 2016.

[57] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-Excitation Networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018.

[58] Vivien Sainte Fare Garnot, Loic Landrieu, and Nesrine Chehata. Multi-Modal Temporal Attention Models for Crop Mapping from Satellite Time Series. *arXiv preprint arXiv:2112.07558*, December 2021. arXiv:2112.07558 [cs, eess] version: 1.

Appendix A

Supplementary Figures

A.1 MineSegSAT

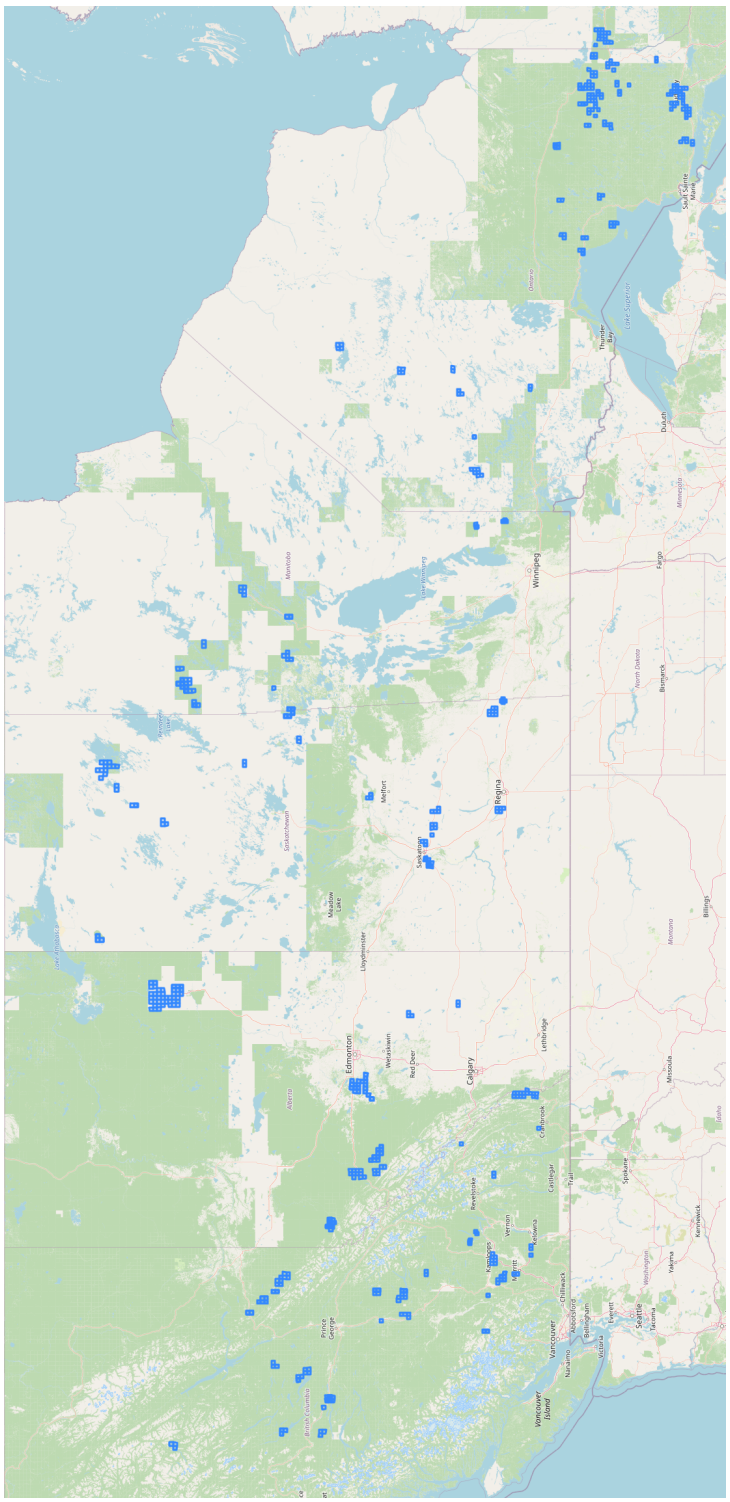


Figure A.1: Mineral Extraction Tiles in Canada used in MineSegSAT

A.2 VistaFormer

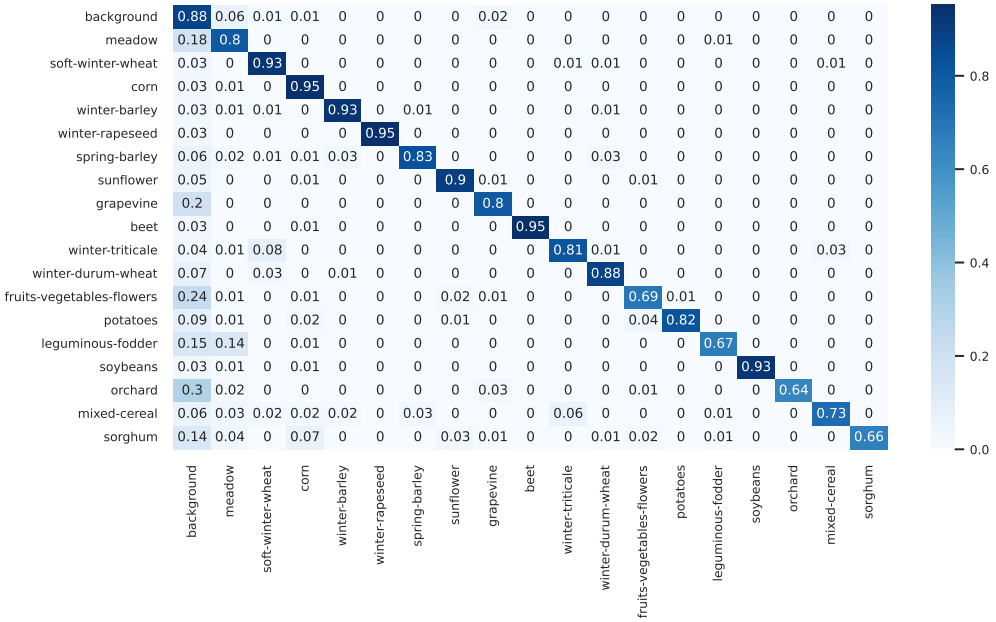


Figure A.2: VistaFormer PASTIS Confusion Matrix

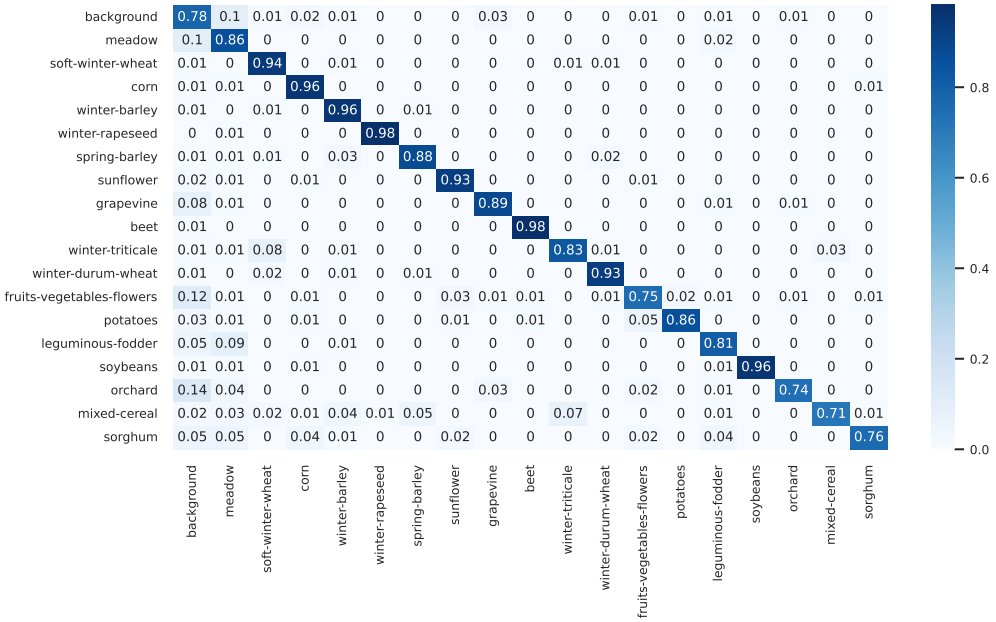


Figure A.3: VistaFormer Neighbourhood PASTIS Confusion Matrix

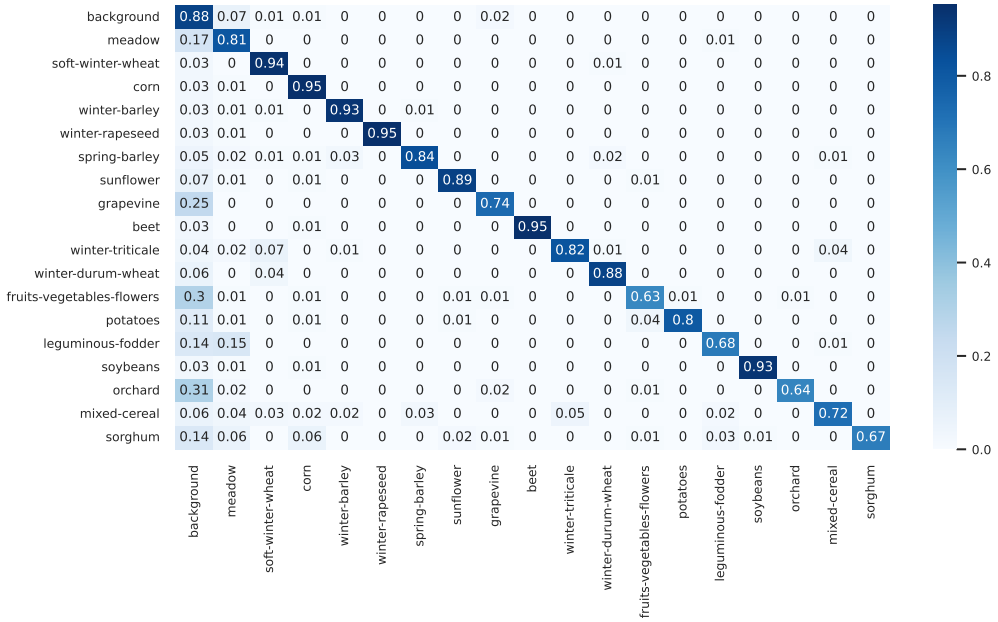


Figure A.4: VistaFormer PASTIS Multi-Input Confusion Matrix

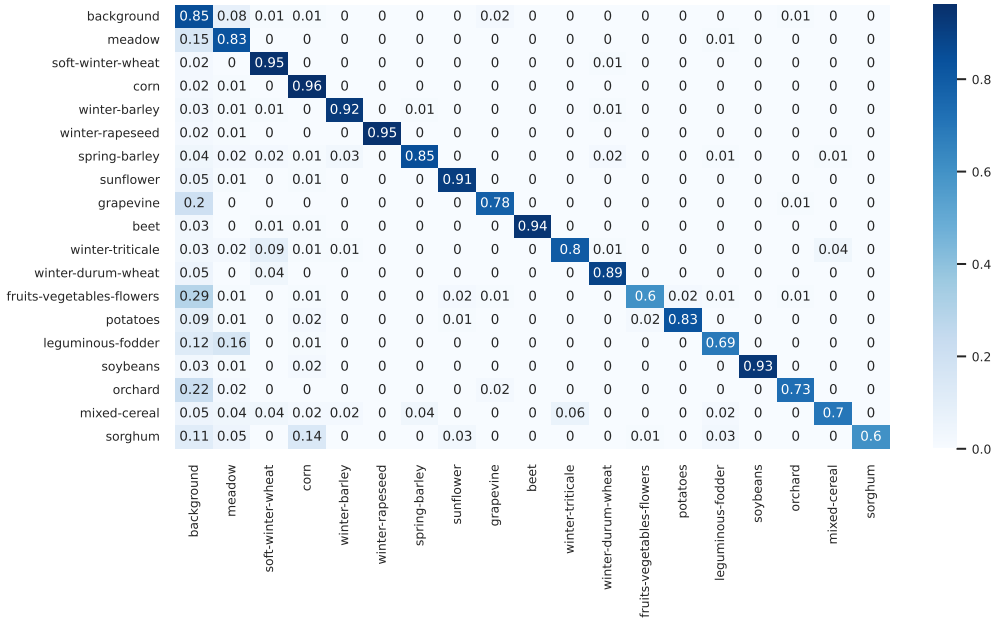


Figure A.5: VistaFormer PASTIS Multi-Input (with Aux-classifier) Confusion Matrix

Appendix B

Supporting Materials

B.1 Code Assets

Code that was created as a part of this thesis is publicly available on GitHub at:

- **MineSegSAT** - https://github.com/macdonaldezra/MineSegSAT
- **VistaFormer** - https://github.com/macdonaldezra/VistaFormer

Code that this paper leveraged considerably for data pre-processing, training, and inference is publicly available on GitHub at:

- **U-TAE** - https://github.com/VSainteuf/utae-paps
- **DeepSatModels** - https://github.com/michaeltrs/DeepSatModels

B.2 Papers Published and Under Preparation

- MacDonald, E.; Jacoby, D.; Coady, Y. MineSegSat: an automated system to evaluate mining disturbed area extents from Sentinel-2 imagery, in *Proceedings of the 5th International Electronic Conference on Remote Sensing*, 7–21 November 2023, MDPI: Basel, Switzerland, doi:10.3390/ECRS2023-16886

- *In Preparation:* MacDonald, E.; Jacoby, D.; Coady, Y. VistaFormer: Simple Vision Transformers for Satellite Image Time Series Segmentation

B.3 Data Assets

Datasets including input samples for MineSegSAT; and trained model weights and training logs for both MineSegSAT and VistaFormer have been made available on Zenodo for download.

- Data that has been produced including trained weights and datasets used for MineSegSAT experiments has been made available at:
  doi:10.5281/zenodo.11236397

- The trained weights and training logs for VistaFormer model experiments presented in the results section of this thesis have been made available at:
  doi:10.5281/zenodo.12667829