Independent University **Bangladesh (IUB)**

IUB Academic Repository

Computer Science and Engineering | Undergraduate Thesis

2025-12

Semi Supervised LULC segmentation with state of the art encoder IJEPA

Jit, Tahmid Hossain

Independent University, Bangladesh

https://ar.iub.edu.bd/handle/11348/1250

Downloaded from IUB Academic Repository



Semi Supervised LULC segmentation with state of the art encoder IJEPA (Image based Joint Embedding Predictive Architecture)

December 2025

Prepared by:

Tahmid Hossain Jit
ID: 2020345
Department of Computer Science
Independent University, Bangladesh

Supervised by:

AKM Mahbubur Rahman
Associate Professor
Department of Computer Science and Engineering

Evaluation Committee

Supervision Panel

Supervisor Dr. AKM Mahbubur Rahman	Co-supervisor/Internal Examiner

Examiners

Internal Examiner 1	Internal Examiner 2

External Examiner (Mr. Asif Md. Akhlak, Head of Operations (Engineering), Fronture Technologies Limited)

Office Use

Dr. AKM Mahbubur Rahman Program Coordinator	Dr. Saadia Binte Alam Head of the Department

Prof. M Arshad Momen
Director, Graduate Studies, Research and Industry Relations,

Md. Shahajada Masud Anowarul Haque, Librarian

Acknowledgement

First and foremost, I would like to express my sincere gratitude to my respected professors, Dr. Amin Ahsan Ali and Dr. AKM Mahbubur Rahman of Independent University, Bangladesh. At the beginning of this journey, when I struggled with almost every aspect of research and deep learning, Professor Amin Ahsan Ali guided me with patience and care. He provided me with research papers, advised me on which courses to take to strengthen my foundations, and reviewed my weekly progress. His direction during these early stages helped me develop the confidence and clarity needed to move forward. He later introduced me to the LULC group at CCDS and to Professor AKM Mahbubur Rahman.

I am profoundly grateful to Professor AKM Mahbubur Rahman, my supervisor, whose guidance, expertise, and encouragement have been central to the completion of this thesis. He taught me most of what I now understand about research . In the beginning, I made many mistakes and found both deep-learning implementation and academic literature overwhelming. But my supervisor always wanted dedication over perfection and that kept me motivated . Under his supervision, I was introduced to the research workflow of the LULC group and gained invaluable hands-on experience.

Lastly I would like to extend my heartfelt thanks to all the Research Assistants of CCDS from whom i learned a lot, especially those in the LULC group. Even still to this day there is always some new technology or some new method or some new approach in coding to learn from them. Despite my constant questions, they were always eager to teach, and somehow always have an answer to a problems I face.

Contents

1 Introduction **6**
- 1.1 Problem Statement . . . 6
- 1.2 Contribution Of The Project . . . 7

2 Literature Review **8**
- 2.1 Existing Datasets and Applied Models . . . 8
  - 2.1.1 Public LULC Datasets . . . 8
- 2.2 Issues with Existing Datasets . . . 9
  - 2.2.1 Domain Mismatch . . . 9
  - 2.2.2 Inconsistent Class Definitions . . . 10
  - 2.2.3 Limited Resolution or Coverage . . . 10
  - 2.2.4 Annotation Cost and Errors . . . 10
- 2.3 Related work on Self-Supervised Learning and Vision Transformers . . . 11
  - 2.3.1 SSL in Remote Sensing . . . 11
  - 2.3.2 ViT in Remote Sensing . . . 12
  - 2.3.3 Why Choose I-JEPA . . . 12

3 Introduction to BingRGB and unlabeled Dataset **13**
- 3.1 Bing RGB Dataset . . . 13
  - 3.1.1 Source Imagery and Acquisition . . . 13
  - 3.1.2 Patch Extraction Strategy . . . 14
  - 3.1.3 Ground truth preparation (RGB $\rightarrow$ grayscale masks) . . . 16
  - 3.1.4 Mask Encoding Pipeline . . . 18
  - 3.1.5 Annotation Quality and Limitations . . . 18
- 3.2 FullBD Unlabeled Dataset . . . 18
  - 3.2.1 Source Imagery and Acquisition . . . 19
  - 3.2.2 Patch Extraction Pipeline . . . 19

4 Methodology **19**
- 4.1 Overview and Architectures . . . 19
  - 4.1.1 Segmenter ViT-16 . . . 19
  - 4.1.2 I-JEPA Pretraining Framework . . . 21
- 4.2 Integrating the I-JEPA Encoder into Segmenter . . . 24
- 4.3 Loss Functions . . . 25
  - 4.3.1 SoftMax . . . 25
  - 4.3.2 CrossEntropy Loss . . . 25
  - 4.3.3 L2 Loss . . . 25
  - 4.3.4 IoU Metrics . . . 26
  - 4.3.5 F1 Accuracy . . . 26

5 Experimental Design **27**
- 5.1 Experimental Settings . . . 27
- 5.2 Dataset Split and Preprocessing . . . 27

5.3 I-JEPA Pretraining on the FullBD Unlabeled Dataset 27
5.3.1 Subset Sampling Strategy 27
5.4 Normalization and Train 28
5.5 Supervised Training 28
5.6 Hardware and Limitation 28

6 Result and Analysis **28**
6.1 Loss Curve 28
6.1.1 Segmenter 28
6.1.2 Segmenter+Ijepa 30
6.2 Class-wise Performance 31
6.3 Predicted Patches 32
6.4 Fail Case 33

7 Conclusion and Future Works **33**
7.1 Conclusion 33
7.2 Future Works 34

1 Introduction

Land-Use/Land-Cover (LULC) mapping from satellite imagery is a fundamental component of environmental monitoring, agricultural planning, disaster management, and urban development assessment. With the rapid growth of remote sensing technology, large volumes of high-resolution satellite images have become readily available for most countries. However, transforming these raw images into meaningful semantic maps requires accurate semantic segmentation models capable of consistently identifying classes such as farmland, water bodies, forested regions, built-up areas, and meadow zones. Also, it has been seen that specifically for satellite imagery data from one climate zone tend to suffer performance degradation when applied to regions with different climate/environment characteristics, indicating that segmentation models generalize better when trained and tested on the same regional data [21] [18]. Experiments on the LoveDA[34] data set show that models trained in one geographic style obtain significantly better segmentation performance when evaluated in regions with similar urban characteristics than when transferred to rural scenes with different land-cover distributions and visual appearances. Hence, Region specific data are required for good segmentation results.

Initially, semantic Segmentation tasks were performed using Fully Connected Convolutional Neural Networks [23], then after advancement, different techniques were applied to make it even better [28] [4]. Recent advances in deep learning—especially Vision Transformers (ViT) and encoder-decoder architectures—have significantly improved segmentation performance. In particular, the Segmenter [30] architecture, which leverages transformer-based encoders and a mask transformer decoder, has demonstrated strong results across several benchmarks. However, it always may not be the case that VIT would outperform FCN [11] as seen from some datasets[26], high-performing segmentation models typically rely heavily on supervised pretraining using ImageNet [10] or other large labeled datasets. For satellite imagery, annotated datasets at this scale are rarely available due to the cost and expertise required for manual labeling.

Self-supervised representation learning has emerged as a solution to this challenge. The old representation learning method VAE [19] and MAE [15] which reconstructed raw pixels, which forces the model to learn textures, lighting noise, sensor noise,background artifacts which created problems such as blurry reconstructions, posterior collapse, weak spatial reasoning, etc. Among the newest approaches, the Image-based Joint Embedding Predictive Architecture (I-JEPA) [1] trains vision models without requiring human labels by predicting high-level representations of masked image regions. Because it learns from images alone, I-JEPA is a promising candidate for pretraining encoders that can later be adapted to downstream remote-sensing tasks such as segmentation.

1.1 Problem Statement

This thesis investigates whether self-supervised pretraining using I-JEPA can improve semantic segmentation performance on a high-resolution satellite dataset, specifically the Bing RGB LULC dataset.Hence I-JEPA is trained under the whole bangladesh unlabeled dataset Problem is that most standard pretraining schemes—especially ImageNet-based supervised pretraining—do not align well with satellite imagery, which differs in texture, distribution,

and object structure . As a result, segmentation models initialized with generic supervised weights may fail to generalize optimally to remote-sensing scenes. Hence another dataset is introduced which contains the unlabeled satellite imagery of the whole Bangladesh, as the data is taken using the same satellite and of the same region , the generalization would be better.

The project focuses on the following core problems:

1. Can a self-supervised encoder trained with I-JEPA learn meaningful representations from Bangladesh satellite imagery without labels?

2. Does integrating this pretrained encoder into the Segmenter architecture yield better segmentation results than using Segmenter alone (vanilla Segmenter)?

3. How does this hybrid I-JEPA + Segmenter model perform compared to traditional segmentation models such as DeepLabV3Plus, U-Net, and HRNet on the same dataset?

4. Are the performance improvements sufficiently consistent across all six LULC classes? (Farmland , Water , Forest , BuiltUp , Meadow)

These questions arise from both practical and academic needs: improving segmentation quality for real-world applications and studying the usefulness of modern self-supervised vision models in remote sensing.

1.2 Contribution Of The Project

The outcome of the project is a comprehensive comparative study that evaluates three major components:

1. Implementation of the self-supervised I-JEPA encoder for training directly on the unlabeled Full Bangladesh satellite dataset.

2. Implementation of Vanilla segmentation models (Vanilla Segmenter, DeepLabV3, U-Net, HRNet) for training under the same data splits and evaluation settings with BingRGB dataset in supervised settings.

3. Implementation of the intregation of the Segmenter-based decoder (mask-transformer) with the I-JEPA pretrained encoder in Semi-Supervised settings. (Pretrained with I-JEPA on unlabeled Full BD and finetuned with segmenter on BingRGB.)

The experiments show that the I-JEPA + Segmenter model provides noticeable improvements in certain classes and consistency across the dataset. However, the improvements are not yet strong enough for publication-level results, primarily due to limited VRAM, model capacity constraints (ViT-Base instead of ViT-Large/Huge), and dataset complexity. Nevertheless, the evaluation demonstrates the viability of self-supervised pretraining in remote sensing and provides a framework for future work involving larger models, more extensive pretraining, and semi-supervised or weakly supervised extensions.

Overall, the thesis establishes a foundation for future research on improving segmentation for satellite imagery using modern self-supervised learning techniques.

2 Literature Review

2.1 Existing Datasets and Applied Models

A wide range of Land-Use/Land-Cover (LULC) datasets has been introduced over the past decade, driven by the growing need for detailed environmental and urban monitoring. Many of these datasets vary significantly in terms of spatial resolution, geographic coverage, number of classes, annotation quality, and intended application. Commonly referenced datasets include:

2.1.1 Public LULC Datasets

1. EuroSAT [17] (RGB and multi-spectral): Based on Sentinel-2 imagery with 13 spectral bands. Provides 27,000 images covering 10 classes such as annual crop, permanent crop, residential, and river. As the dataset is classification based, it is not annotated pixelwise like segmentation datasets.

2. DeepGlobe [9] Land Cover Challenge Dataset: Taken with DigitalGlobe Vivid+ Satellite from Thailand , Indonesia , India , USA which Contains high-resolution tiles with a Ground Sampling Distance of 0.5m, with a resolution of 2448x2448 annotated with 8 classes such as agriculture, rangeland, forest, water, urban, barren . Frequently used for benchmarking segmentation architectures.

3. SpaceNet [32] taken with DigitalGlobe/Maxar Satellites Focused primarily on building footprints and road extraction. however, several subsets also support multi-class segmentation tasks. The Data were taken from LasVegas , Paris , Shanghai , Khartoum etc.

4. LoveDA [34] Urban vs. rural segmentation across three cities (Nanjing , Changzhou , Wuhan) from China with very detailed pixel-level masks.

5. CitySkapes [8] A high-resolution urban street scene segmentation dataset captured from car-mounted cameras in 50 European cities. It contains 5,000 finely annotated images covering 19 classes, including buildings, vegetation, roads, cars, pedestrians, and sky. Although it is not satellite imagery, Cityscapes is one of the most influential datasets in semantic segmentation research, especially for developing and validating deep architectures.

Although these datasets are widely used in research, most of them do not capture the unique textures, colors, and land patterns of Bangladesh's geography. For example, EuroSAT, CitySkapes and DeepGlobe contain European or North American imagery, which differs significantly from South Asian terrain. LoveDA has Asian terrain but even their environments differ from that of Bangladesh.

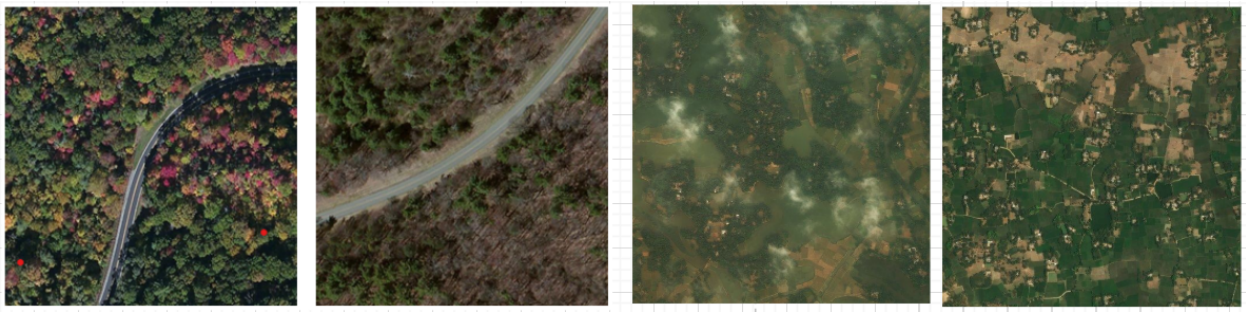
2.2 Issues with Existing Datasets

Despite the availability of numerous Land-Use/Land-Cover (LULC) datasets in remote sensing research, several limitations restrict their suitability for training high-performance segmentation models—especially when the target domain involves high-resolution imagery of Bangladesh. These issues contribute significantly to domain shift and poor generalization.

2.2.1 Domain Mismatch

Most widely used LULC datasets are derived from geographic regions that differ substantially in:

- Vegetation patterns and crop structures European or North American farmland layouts, seasonal vegetation colors, and canopy textures differ from Bangladesh's agricultural landscape.[21] [27]



- Urban morphology Differences in building density, rooftop materials, informal settlements, and road networks introduce large structural discrepancies.[18] [34]



- Hydrological characteristics Water bodies in Bangladesh vary in turbidity, sediment load, and river morphology, unlike clearer or narrower water features seen in Western datasets. [27] [2]



Because segmentation models are highly sensitive to texture and contextual patterns, models pretrained on these datasets exhibit weaker performance when transferred to Bangladeshi satellite scenes.

2.2.2 Inconsistent Class Definitions

Different datasets define land-cover classes in inconsistent and sometimes ambiguous ways. Examples include:

- Merging or splitting agricultural categories (e.g., "meadow", "grassland", "farmland").
- Varying notions of "built-up" areas (industrial, residential, mixed).
- Some datasets include background labels; others do not.

This inconsistency makes cross-dataset training or benchmarking difficult and prevents direct comparison of model performance. For Bangladesh, no standard high-resolution 6-class segmentation dataset existed prior to the BingRGB dataset used in this work, which reinforces the need to design a dataset tailored to the region.

2.2.3 Limited Resolution or Coverage

Main problem is that publicly mostly low resolution segmentation datasets are available which has very large datasets. But for LULC tasks high-resolution data is required and High-resolution segmentation datasets that are available are usually very small in quantity. In spatial coverage—capturing they only have a few cities or small regions from multiple cities around the world.In OpenEarthMap [35] many regions around the world were selected not a specific country. These datasets often include only a few thousand labeled samples, which restricts the training of capacity-heavy models such as Vision Transformers and modern decoders (Segmenter [30], SegFormer [36], HRNet [33]).

This limitation directly motivates assembling large unlabeled datasets (such as FullBD) to support self-supervised learning.

2.2.4 Annotation Cost and Errors

- Labels often contain boundary noise, especially around mixed vegetation and water edges.

- Manual labeling may suffer from inter-annotator variability.
- Shadows, cloud artifacts, and seasonal changes introduce labeling uncertainty.
- Datasets from different sources use different annotation protocols, reducing compatibility.Can be in terms of color coding, class definitions.

2.3 Related work on Self-Supervised Learning and Vision Transformers

In recent years, self-supervised learning (SSL) and transformer-based architectures have significantly influenced computer vision and remote-sensing research. SSL reduces dependence on large labeled datasets, while Vision Transformers (ViTs) offer improved capacity for modeling spatial relationships across large receptive fields. This section reviews prior work relevant to the motivation behind integrating I-JEPA with Segmenter for LULC segmentation.

2.3.1 SSL in Remote Sensing

Self-supervised learning has gained popularity in remote sensing due to the scarcity of large-scale pixel-level annotations. Early SSL approaches adapted contrastive learning methods—such as MoCo [16], SimCLR [5], and BYOL [14] to satellite imagery by enforcing similarity between augmented views of the same patch. These approaches helped models learn robust feature embeddings without supervision.

- SeCo [24], which uses season-based contrastive learning to capture seasonal variations in satellite imagery.
- SatMAE [7], one of the first adaptations of Masked Autoencoders (MAE) for remote sensing, demonstrating significant improvements in downstream classification and segmentation tasks.
- RemoteCLIP [20] and related large-scale vision–language pretraining approaches, which leverage textual metadata for weak supervision.
- DINO [37] and iBOT[38]-based adaptations that show strong multi-modal alignment with Sentinel-1/Sentinel-2 imagery.

These works collectively demonstrate that SSL can outperform ImageNet-pretrained baselines when the model is pre-trained directly on satellite domain data. However, most SSL methods reconstruct pixels or rely on augmentation-heavy contrastive objectives, which may not fully capture structural information in high-resolution imagery.

2.3.2 ViT in Remote Sensing

Vision Transformers (ViTs) [12] were originally introduced for natural image classification, but their ability to model long-range dependencies has attracted significant interest in remote sensing. Several transformer-based architectures have since been adapted for segmentation and classification tasks:

- Swin Transformer [22] and its variants have been used for building extraction, road segmentation, and LULC classification due to their hierarchical window-based attention.
- SegFormer [36] demonstrated state-of-the-art performance in remote-sensing segmentation with a lightweight MLP decoder.
- Segmenter [30], used in this thesis, employs pure ViT encoders and a mask transformer decoder, offering strong performance on semantic segmentation benchmarks.
- MAE[15]-pretrained ViTs have been applied to remote-sensing classification and segmentation with notable success, as seen in works such as SatMAE[7] and other ViT-base/large adaptations.

Transformers outperform CNNs (U-Net, DeepLab) in many cases due to their Global receptive field, Ability to learn texture-independent features, Better modeling of large homogeneous areas (common in satellite images).Still If it will outperform or not also depends on the dataset itself and also the encoder parameter size. Generally larger ViTs performs better than their smaller counterpart. However, transformers require large-scale pretraining—often unavailable for region-specific satellite imagery—making self-supervised methods increasingly attractive.

2.3.3 Why Choose I-JEPA

I-JEPA (Image-based Joint Embedding Predictive Architecture) [1] represents a shift from pixel-based reconstruction (as used in MAE) to predicting abstract latent representations of masked regions. This approach offers several advantages that make it suitable for remote-sensing applications:

a) **Prediction in latent space, not pixel space**

Pixel reconstruction biases models toward low-level textures, which can be highly variable in satellite imagery. I-JEPA predicts embeddings, focusing on structure and semantics rather than exact pixel values.

b) **More stable and compute-efficient pretraining**

I-JEPA avoids decoder-heavy architectures, reducing training cost. This is beneficial when pretraining on large unlabeled datasets like unlabeled FullBD.

c) **Better alignment with segmentation tasks**

Segmentation requires coherent region-level understanding. I-JEPA's latent predictions encourage global spatial consistency, which is needed for distinguishing classes such as forest vs. meadow or built-up vs. farmland.

d) **Demonstrated effectiveness on non-natural images**

The original I-JEPA paper reported strong results in transfer learning, particularly on tasks requiring structural reasoning—similar to the demands of satellite segmentation.

e) **Complementary with transformer decoders**

Because Segmenter already uses a ViT encoder, replacing the encoder with an I-JEPA–pretrained one creates a natural and efficient architecture pipeline.

Thus, I-JEPA provides a promising mechanism for learning region-specific representations from unlabeled Bangladesh imagery.

3 Introduction to BingRGB and unlabeled Dataset

3.1 Bing RGB Dataset

The BingRGB dataset forms the core supervised component of this research and serves as the primary benchmark for evaluating semantic segmentation performance across U-Net, DeepLabV3+, HRNet, Segmenter, and the proposed I-JEPA-pretrained Segmenter. This section provides a comprehensive overview of the dataset creation process, beginning from raw satellite acquisition to patch extraction, class consolidation, and mask preparation.

3.1.1 Source Imagery and Acquisition

The raw satellite imagery was obtained from Bing Maps Aerial imagery, using Zoom Level 17, which corresponds to an approximate ground sampling distance of 2.22 meters per pixel. At this resolution, major land-cover features such as agricultural fields, water bodies, urban structures, and forested regions are visually distinguishable and suitable for pixel-level semantic segmentation tasks. Imagery was initially downloaded and stored as GeoTIFF files to retain: Geolocation metadata, Projection information, Band ordering and high-precision RGB values Storing the data in TIFF format allowed accurate visualization, geoprocessing, and alignment with vector layers inside QGIS[13] .

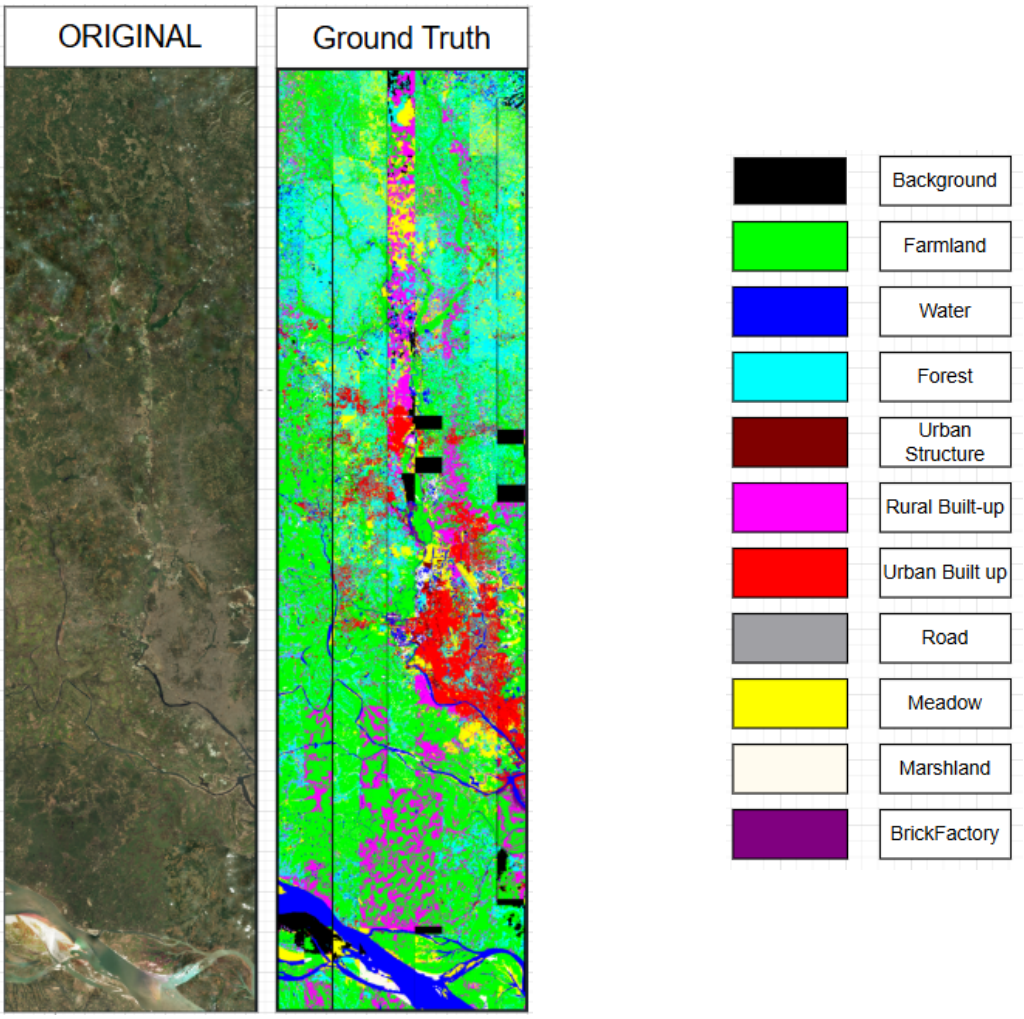


Figure 1: Source image and corresponding color-coded ground truth.

Figure-1 shows the original downloaded image along with the annotated ground truth and beside is the color code on which color is used to represent which class.

The collected BingRGB dataset covers the entirety of the Dhaka Division, an administrative region in central Bangladesh. The coverage area spans approximately: 117 km (east–west), 124 km (north–south), Resulting in a raster dimension of approximately 48,906 × 47,256 pixels. This coverage includes diverse urban and rural environments such as metropolitan Dhaka, peri-urban settlements, agricultural fields, water channels, floodplains, meadows, and forest patches.

3.1.2 Patch Extraction Strategy

To convert the large GeoTIFF mosaic into manageable units for model training, a detailed patch-generation pipeline was developed. Each RGB image was cropped into 513 × 513 patches, a resolution selected to balance GPU memory constraints with the need for spatial context.

1. **Training Set Patch Extraction**

A Patch size of 513 × 513 with Overlap: 75 percent, (achieved using stride 128). The idea is that it would provide dense coverage, increase training diversity and ensure class continuity across neighboring patches This resulted in 35,857 training patches, all sampled from the Dhaka Division except the region reserved for testing.

2. **Test Set Patch Extraction** To ensure a fair evaluation, the test region was cropped out of the main mosaic before generating training patches, preventing any spatial overlap between train and test sets. A Patch size: 513 × 513 with no overlap (achieved using stride 513). The Region this set covers is Only the Dhaka metropolitan urban zone hence the resultant patches are a Total 286 patches.This strict geographic separation ensures that the test set truly reflects generalization to unseen areas.

For both Train and test patches During dataset preparation, regions annotated as background (class 0, represented by black pixels) were explicitly enforced in the original imagery. Specifically, wherever the annotation mask indicated background, the corresponding pixels in the RGB image were overwritten with black.

The motivation behind this design choice was to prevent the model from learning spurious associations with black pixels that might occur randomly in the imagery. By constraining black to appear only in regions explicitly annotated as background, the segmentation network learns a consistent mapping: a completely black pixel corresponds exclusively to the background class. This reduces the likelihood of erroneous background predictions in non-background regions and improves the robustness of the model's class discrimination.

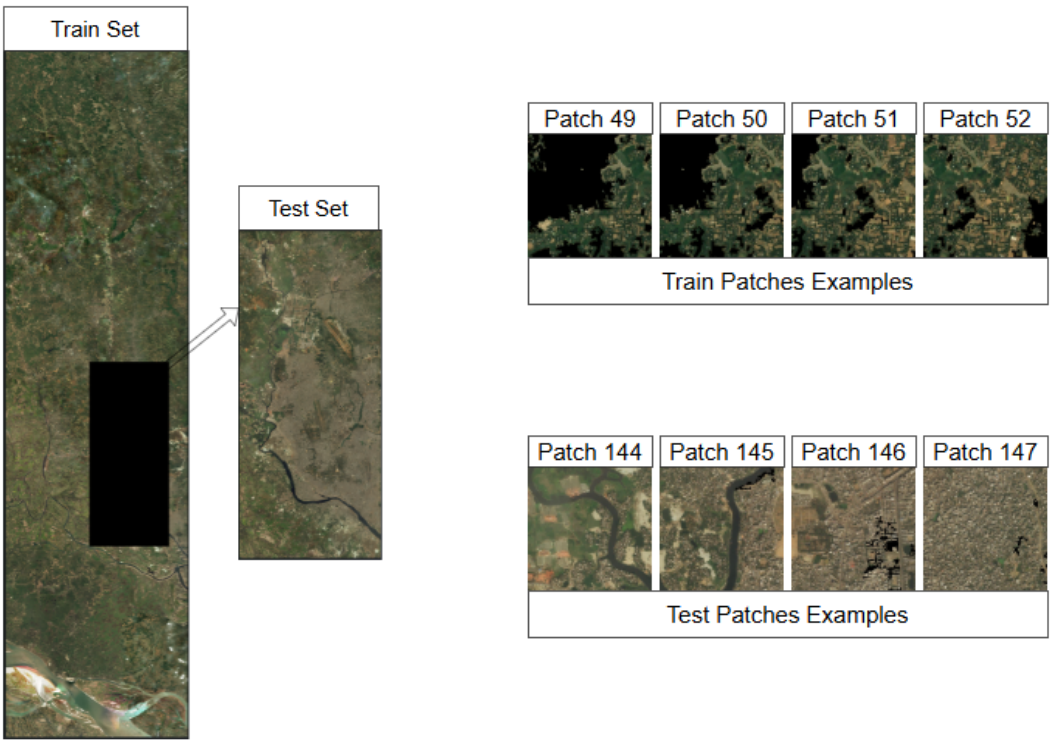


Figure 2: Train Set vs Test Set , Train Patches vs Test Patches

Figure-2 shows that the train set is the whole dhaka division as our original set only the dhaka metropolitan city is cut out for test set. And next to it we can see that training patches has overlapping tiles while test patches does not. Also we can see that regions which was catagorized as background were also imposed on original image

3.1.3 Ground truth preparation (RGB → grayscale masks)

The annotated ground truth which consisted of 11 LULC classes were exactly patched according to their respective original image counterparts.The 11 Classes included detailed urban categories and multiple vegetation types. However, for practical segmentation purposes and to reduce annotation noise, these were also merged into six final classes:

Final Class	Value	Original Classes Merged	Color Example
Background	0	Black	(0,0,0)
Farmland	1	Farmland	(0,255,0)
Water	2	Water	(0,0,255)
Forest	3	Forest	(0,255,255)
Built-up	4	Urban built-up, rural built-up, urban structure, brick factory, road	(128,0,0), (255,0,0), (160,160,164), (128,0,128), (255,0,255)
Meadow	5	Meadow, marshland	(255,255,0), (255,251,240)

Table 1: Final class mapping, merged categories, and corresponding RGB color values.

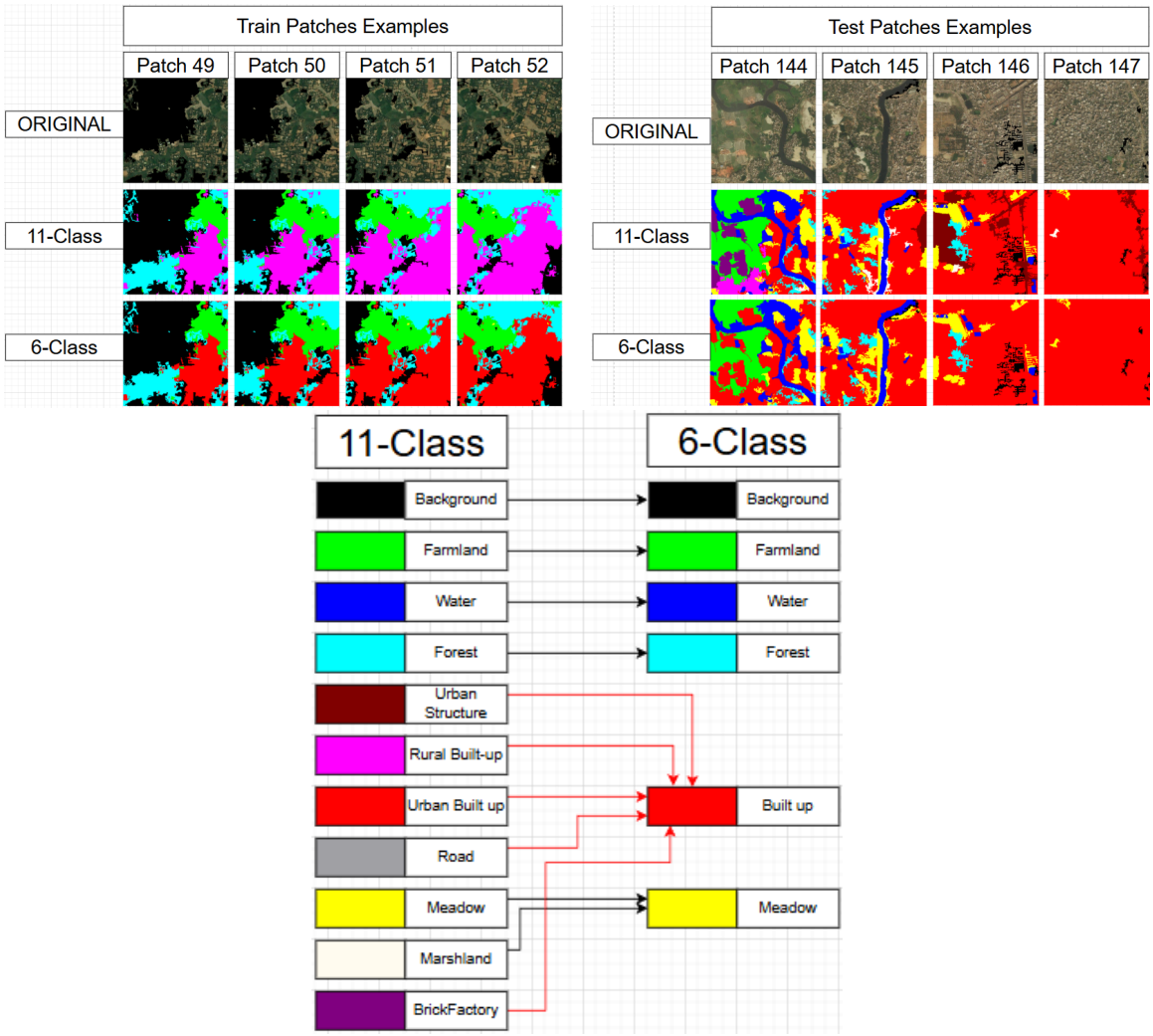


Figure 3: Class-wise Mapping and Class-wise Patches Example

Figure-3 shows that the classmapping of 11 Class to 6 Class and also it shows some examples of patches with both 11 Class and 6 Class.

3.1.4 Mask Encoding Pipeline

The annotated data was originally provided as RGB-colored masks, with each land-cover category represented by a unique RGB triplet. For model training, these were converted into grayscale integer masks, where each pixel contains a class index (0–5).A script was created whose goal is to:

1. Read each RGB mask
2. Match pixel colors to their class label using the defined color map
3. Generates a 2D array of integer values (0–5) and save the processed masks to train/gts/... and test/gts/...

3.1.5 Annotation Quality and Limitations

Although the original annotations were manually created or sourced from prior LULC datasets, some class boundaries were imperfect or inconsistent. However, these inconsistencies were significantly reduced after merging the classes from 11 to 6.

Remaining limitations include:

- Minor misalignment between RGB imagery and masks in areas with dense buildings
- Occasional disagreement between "meadow" and "forest" boundaries
- Built-up areas with atypical colors (e.g., tin rooftops) causing slight ambiguity

Despite these issues, the dataset is clean, diverse, and sufficiently large to support transformer-based segmentation models.

3.2 FullBD Unlabeled Dataset

The FullBD Unlabeled Dataset serves as the large-scale, self-supervision corpus used to pretrain the I-JEPA encoder. Unlike the BingRGB dataset—where pixel-level ground truth masks are available—the FullBD dataset contains only raw RGB satellite imagery without annotations. Its purpose is exclusively to support representation learning across the diverse geographic, climatic, and structural variations found across Bangladesh.

This section describes the acquisition, structure, patch extraction, preprocessing characteristics, and the underlying motivation for using this dataset for self-supervised pretraining.

3.2.1 Source Imagery and Acquisition

The FullBD dataset was constructed using high-resolution Bing Maps Aerial imagery, obtained at Zoom Level 17, corresponding to approximately 2.22 meters per pixel identical to the BingRGB dataset. The dataset originally consisted of 64 large mosaic images, each representing one of the 64 districts of Bangladesh. Although the internal format of the raw files may include both GeoTIFF and PNG variants, the available working dataset consists of PNG images, which were sufficient for patch extraction and pretraining. Because the imagery was downloaded as rectangular bounding boxes around districts, areas lying outside the true district boundaries but within the rectangular tile appear as black (RGB value 0,0,0). This masking is important to note because it affects the distribution of pixel intensities in the dataset.

3.2.2 Patch Extraction Pipeline

To construct a large-scale unlabeled training set for I-JEPA, each of the 64 district mosaics was patched using the same extraction code and parameters used for BingRGB. Patch Specifications are exactly the same as bingRGB, however this time we dont have annotation. Since the code was executed in a loop of 64 (once for each district), the final consolidated data set contains 123,496 unlabeled patches. Stored in a single flat directory (no district sub folders)

This large number of examples supports stable pretraining for models like I-JEPA, which rely on high-volume data for effective representation learning. The main Reason For Choosing This Dataset was that it had many images and as it has Bangladesh as a whole it is very diverse consisting of all different classes which would create a good Generalized representation .

4 Methodology

4.1 Overview and Architectures

4.1.1 Segmenter ViT-16

Segmenter [30], introduced as a transformer-based semantic segmentation model that replaces CNN backbones and decoder modules with a fully attention-driven design. Unlike CNN-based segmentation architectures such as U-Net or DeepLabV3+, Segmenter formulates segmentation as a sequence-to-sequence prediction task in which image patches are encoded via a Vision Transformer (ViT), and a transformer-based decoder predicts class assignments for each patch token. The core idea is to leverage the global reasoning capability of self-attention to model long-range spatial interactions, which is particularly advantageous in remote sensing, where land-use patterns often contain large, structured regions with complex contextual dependencies.

- **Patch Tokenization and Linear Embedding** Segmenter begins by partitioning the input image into fixed-size non-overlapping patches (e.g., 16×16 pixels for ViT-Base). Each patch is flattened and projected through a learnable linear embedding

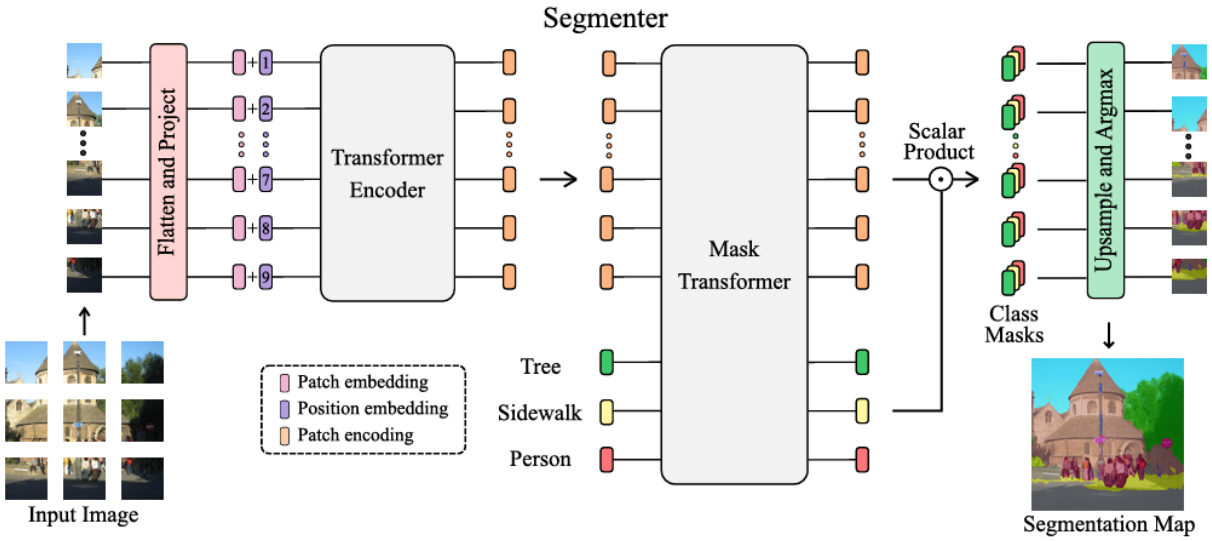


Figure 4: Segmenter Architecture

layer to form a sequence of patch tokens: $x = \{x_1, x_2, \ldots, x_N\}$ here N is the number of patches. A 2D positional embedding is added to preserve the spatial information that transformers do not inherently capture.

- **ViT Encoder: Global Attention Over All Patches** The encoded sequence is passed through a standard Vision Transformer [12]. The ViT consists of: Multi-head self-attention layers,Feed-forward MLP blocks,Layer normalization,Residual connections. The encoder models global context across all patches and outputs a sequence of contextualized patch representations which we can call patch encoding. The output of the encoder is a sequence $Z = \{z_1, z_2, \ldots, z_N\}$. Here Z is the patch encoding.

- **Mask Transformer Decoder**

  Unlike many segmentation models that decode features spatially, Segmenter decodes class masks directly from attention interactions between class tokens and patch tokens.

  Each class k is represented by a learnable vector c_k, which acts as a query that seeks out patch tokens belonging to that class. These class embeddings accumulate semantic information about each category.

  The decoder performs attention between the class embeddings and the ViT patch embeddings. Each class embedding attends to all patch features and Patch features attend back to class embeddings. Through these bidirectional attention operations, the model determines which patches correspond to which classes.

- **Scalar Product Mask Computation**

  The mask transformer produces two structured sets of vectors: class embeddings and per-patch embeddings. Segmenter computes segmentation logits using a scalar product (dot product) between each class embedding and each encoded patch representation. The scalar product measures the similarity between the class representation and the

patch token, effectively scoring how likely that patch belongs to a given class. This process transforms the transformer outputs into class masks—coarse spatial grids indicating class presence at the patch level.

- **Upsampling and Final Segmentation Map**

  Since the predictions are produced at the patch resolution (e.g., $H/16 \times W/16$), Segmenter performs an upsampling step to return to the original image resolution. The upsampling is typically bilinear.

  After upsampling, an argmax is applied across class channels to assign each pixel to the class with highest similarity score, producing the final segmentation map as shown at the far right of the Figure 4.

4.1.2 I-JEPA Pretraining Framework

I-JEPA (Image-based Joint Embedding Predictive Architecture) [1], is a non-generative self-supervised learning framework designed to learn high-level visual representations by predicting abstract embeddings rather than reconstructing raw pixels. Unlike generative masked autoencoders that attempt to reconstruct missing pixel values, I-JEPA predicts the representation of masked image regions in the latent space of a target encoder. This approach leads to more semantic, stable, and scalable training, particularly suited for large unlabeled datasets.The model is fundamentally different from both joint-embedding contrastive architectures and pixel reconstruction architectures, as illustrated in Figure 5.

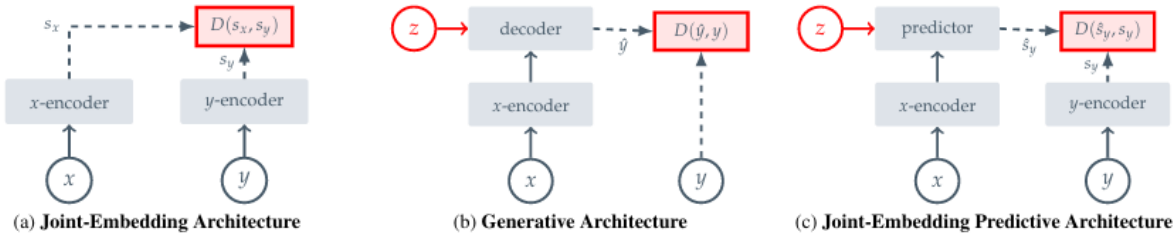


Figure 5: SSL Architectures

a) **Joint-Embedding Architectures**

Methods like BYOL[14] or DINO [3] learn representations by matching the embeddings of two augmented versions of the same image. They operate entirely in embedding space but depend heavily on data augmentation and invariances.So x and y are the images or texts or (image and text) sent to their own encoders Which produces s_x, s_y embeddings respectively at the end. The result passes through the similarty function D that evaluates how compatible the two embeddings by some loss function which are usually cosine similarty inside a constrastive loss (InfoNCE)[25]. The JEA methods usually have inputs of different modalities.

b) **Generative Architectures**

Masked autoencoders (MAE) and similar generative models learn to reconstruct missing pixels. They operate in pixel space and require the decoder to predict raw RGB values, which is computationally expensive and often forces the model to focus on low-level textures.Here x is the augmented version of an image and y is the original version of that image. x passes through the encoder and produces latent representation z , which is then fed into a decoder network which is usually a lightweight transformer that reconstructs the masked images to produce $\hat{y}$ which is used as our prediction. Finally we compute difference between predicted image $\hat{y}$ vs y with a simple L2/MSE Loss.

c) **Joint-Embedding Predictive Architecture (JEPA)**

JEPA , which becomes I-JEPA when applied to images, predicts the latent embedding of missing regions rather than reconstructing pixels. This combines the best of both worlds: Like joint-embedding models, JEPA avoids pixel reconstruction. Like predictive models, JEPA performs a prediction task that encourages semantic understanding. x is the original image, y is the masked version of the exact same image. $x - encoder$ produces the representation of the context region as embedding s_x . $y - encoder$ produces the representation of the target region/ ground truth embedding s_y Both encoders are same backbone. The predictor takes the context representation embedding s_x to output a prediction $\hat{s}_y$. Finally we calculate the L2/MSE loss between $\hat{s}_y$ representation embedding prediction vs s_y ground truth embedding.

- **Context Encoder: Processing Visible Regions**

  I-JEPA begins by partitioning the image into fixed patches, similar to ViT. A block masking strategy is applied, in which certain spatial blocks (e.g., large contiguous rectangular regions) are removed from the input. The remaining visible patches are processed by a Vision Transformer called the context encoder.

  This encoder produces latent representations that summarize the global structure of the visible content.The I-JEPA encoder focuses on producing semantic embeddings that are robust to missing regions, making it more stable and scalable across different image domains.

  Because the encoder sees only part of the image, its representation must rely on global scene understanding, making it well-suited for high-variance remote sensing data where spatial patterns extend across large geographic areas.

- **Target Encoder: Providing Ground-Truth Latent Embeddings**

  In parallel to the context encoder, the target encoder receives the full image, including the masked regions. Importantly, the target encoder is updated using an Exponential Moving Average (EMA) [31] of the context encoder's parameters, similar in spirit to teacher–student architectures such as DINO [3].

  The target encoder produces "true" latent embeddings for the original image patches. These embeddings serve as the ground-truth signals that the predictor network must

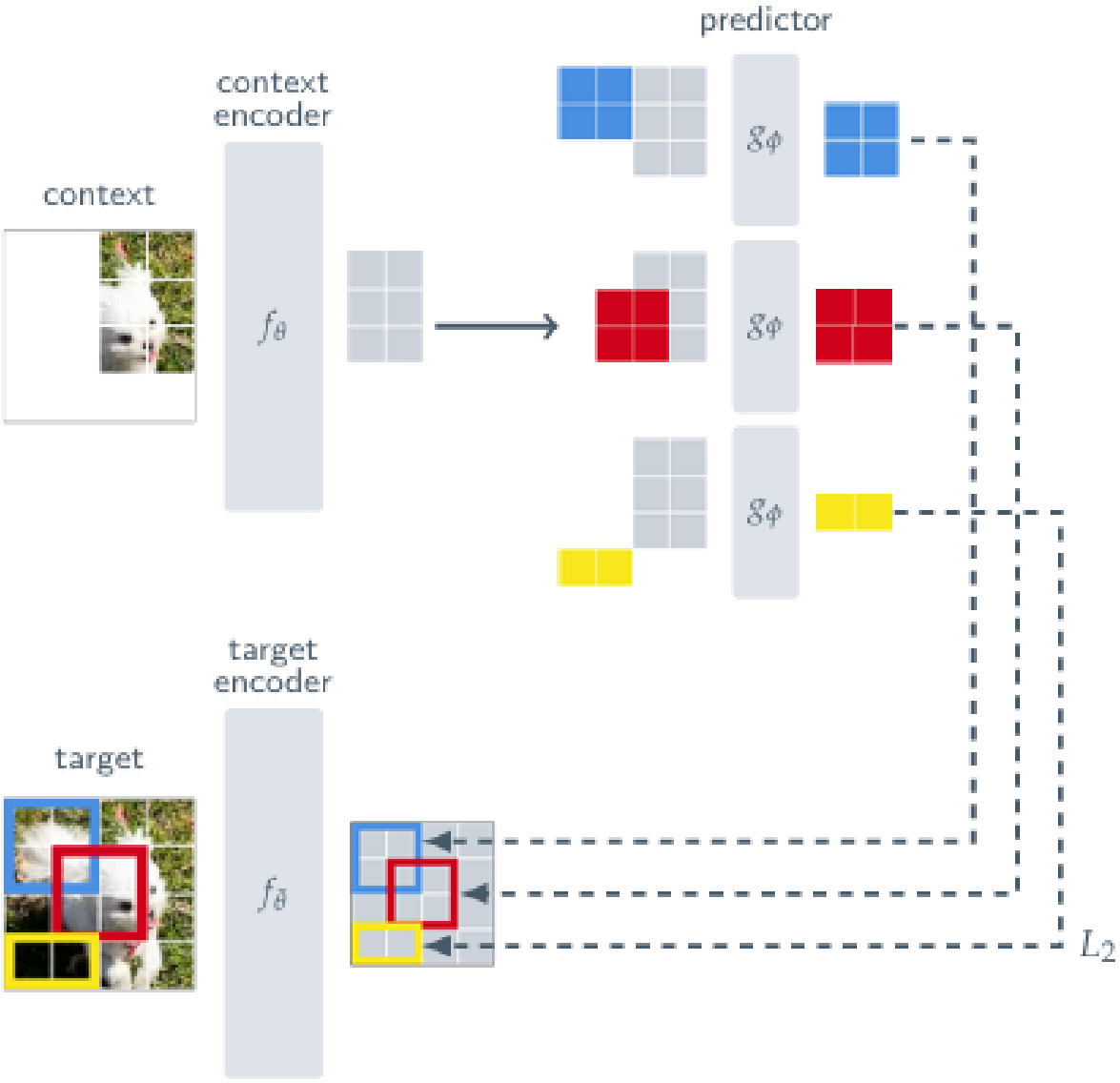


Figure 6: Ijepa Architecture

approximate. Since the target encoder is EMA-updated, it evolves slowly and provides a stable latent space, preventing training collapse.

For masked areas of the image, the target encoder's output represents the semantic embedding of regions the context encoder never sees, enabling the learning of predictive abstraction.

Although the target encoder processes the full image, it outputs one single sequence of patch embeddings, not separate embeddings for masked and visible regions. We denote this sequence as $s_y = \{s_{y1}, \ldots, s_{yN}\}$

where each element corresponds to one patch. During training, I-JEPA simply indexes this unified sequence to extract the embeddings at the positions belonging to masked

blocks. Only these masked-block embeddings are used as the supervision signal. Thus, the target encoder creates a complete latent map, but the training objective uses only the subset corresponding to masked patches.

- Block Masking: Predicting Large Missing Regions

  I-JEPA introduces a blockwise masking strategy specifically designed to avoid the pitfalls of token-wise random masking used in masked autoencoders. Instead of masking tokens at random scattered positions, I-JEPA masks large contiguous blocks of the image. Predicting masked blocks forces the encoder to learn long-range structure, Large mask regions encourage global reasoning, The learned representations generalize better and capture high-level concepts. Masked regions are never fed into the context encoder; only the target encoder observes them. The predictor then learns, indirectly, to approximate their representations.

- **Prediction Mechanism**

  The predictor takes as input the context-encoder embeddings corresponding to visible patches and a learned mask token that identifies which block is being predicted. I-JEPA predicts entire blocks rather than individual patches.For each masked block B_i

$$\hat{s}_y(i) = \{\hat{s}_{y_j}\}_{j \in B_i}$$

  These predictions are in the same latent space as the target encoder's embeddings. The training loss compares the predictor outputs only to the embeddings of the masked blocks extracted from the target encoder. This block-level prediction forces the model to infer semantic properties of large spatial regions rather than reconstructing local textures.

4.2 Integrating the I-JEPA Encoder into Segmenter

After pretraining I-JEPA on the unlabeled FullBD dataset, several ViT encoder checkpoints were produced (latest checkpoint, best-loss checkpoint, and checkpoints saved after every 50 epoch). For each downstream experiment, one of these checkpoints was selected and used to initialize the encoder of the Segmenter model. The remaining components of Segmenter, the mask transformer decoder and the final segmentation head were randomly initialized. During supervised training on the BingRGB dataset, all parameters of the model were updated jointly; the I-JEPA encoder was not frozen. This setup corresponds to full fine-tuning, where I-JEPA provides domain specific initialization but the model learns end-to end on the segmentation task.Since the I-JEPA encoder is fully trainable, gradients update both the pretrained encoder and the randomly initialized decoder layers, allowing the model to adapt its representation to the supervised segmentation task. This integration enables a direct and fair comparison between three key model variants:

- A standard Segmenter initialized from ImageNet-pretrained ViT weights,
- A Segmenter initialized from I-JEPA-pretrained encoder weights,

4.3 Loss Functions

4.3.1 SoftMax

SoftMax is a normalization function that converts raw network outputs (logits) into probabilities for each class. Each probability is between 0 and 1. It is used to predict probability in Cross Entropy Loss

$$\hat{y}_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{1}$$

- z_i the logit (raw output value) predicted by the model for class i at a pixel.
- C the total number of classes
- $\hat{y}_i$ the predicted probability that the pixel belongs to class i

4.3.2 CrossEntropy Loss

Cross-Entropy measures how different the predicted probability distribution is from the true class label. If the model predicts the correct pixel with high probability, loss would be lowest. If the model predicts the wrong pixel with high probability, loss would be highest. Cross Entropy Loss the main loss used in most semantic segmentation tasks. Segmenter uses it.

$$\mathcal{L}_{\text{CE}} = -\frac{1}{N} \sum_{p=1}^{N} \log(\hat{y}_{p,y_p}) \tag{2}$$

- p represents the pixel
- N is the total number of pixels
- y_p The ground-truth (true) class index for pixel p
- $\hat{y_p}$ The predicted probability distribution over all classes for pixel p.

$$\hat{\mathbf{y}}_p = \left[\hat{y}_{p,1}, \hat{y}_{p,2}, \hat{y}_{p,3}, \hat{y}_{p,4}, \hat{y}_{p,5}, \hat{y}_{p,6}\right] \tag{3}$$

4.3.3 L2 Loss

I-JEPA learns representations by predicting missing feature blocks, not pixels. It uses L2 loss to measure how close the predicted representation is to the teacher representation.

$$\mathcal{L}_{\ell_2} = \sum_{k=1}^{D} \left(f_\theta(x)_k - g_\phi(x)_k\right)^2 \tag{4}$$

- $f_\theta(x)$ teacher encoder output (target features)
- $g_\phi(x)$ student predictor output (predicted features)

- D dimension of the representation vector
- x input image patch
- θ, ϕ teacher and student parameters respectively

4.3.4 IoU Metrics

IoU (Intersection over Union) is the main evaluation metric in semantic segmentation. It measures how much the predicted mask region overlaps with the ground-truth region.

$$\text{IoU}_c = \frac{|\text{Prediction}_c \cap \text{GroundTruth}_c|}{|\text{Prediction}_c| + |\text{GroundTruth}_c| - |\text{Prediction}_c \cap \text{GroundTruth}_c|}. \tag{5}$$

- $TP = \text{Prediction}_c \cap \text{GroundTruth}_c$
- $FP = \text{Prediction}_c$ - TP
- $FN = \text{GroundTruth}_c$ - TP

$$\text{IoU}_c = \frac{TP_c}{TP_c + FP_c + FN_c}. \tag{6}$$

- TP_c True Positive pixels correctly predicted as class c
- FP_c False Positive pixels incorrectly predicted as class c
- FN_c False Negative pixels belonging to class c but predicted as something else
- C is total number of classes

$$\text{mIoU} = \frac{1}{C} \sum_{c=1}^{C} \text{IoU}_c. \tag{7}$$

mIoU is the mean IoU over all classes

4.3.5 F1 Accuracy

Precision: Out of all samples that the model predicted positive how many are actually positive?

$$P_c = \frac{TP_c}{TP_c + FP_c}.$$

Recall: Out of all the actual positives in the dataset, how many did the model correctly identify?

$$R_c = \frac{TP_c}{TP_c + FN_c}.$$

F1: is high only if both precision and recall are high

$$F1_c = \frac{2P_cR_c}{P_c + R_c}.$$

5 Experimental Design

5.1 Experimental Settings

This section describes how the self-supervised pretraining (I-JEPA), the supervised model Segmenter and how the semi supervised model (I-Jepa+Segmenter) model were configured and compared. All experiments were run on the same BingRGB dataset split for supervised evaluation, and care was taken to keep the training protocols as comparable as possible across models.

5.2 Dataset Split and Preprocessing

Both the models in this study vanilla Segmenter and I-JEPA+Segmenter use the same BingRGB train/test split:

- **Training set**: 35,857 patches of size 513×513, extracted from the Dhaka Division with 75
- **Test set**: 286 patches of size 513×513, cropped from the Dhaka metropolitan region with no overlap (stride 513) and no spatial intersection with the training patches.

Although the original patches are 513×513, the segmentation pipelines standardize inputs to 512×512 for compatibility with Vision Transformer.

5.3 I-JEPA Pretraining on the FullBD Unlabeled Dataset

I-JEPA is pretrained on the FullBD unlabeled dataset which consists of 123,496 RGB images

5.3.1 Subset Sampling Strategy

To reduce computational cost and explore the effect of data coverage, a subset sampling strategy was implemented during I-JEPA training.So basically at the start of the training a subset ratio would be fixed Sr=(0.2,0.3,0.4,0.5), At each epoch, a random subset of the FullBD patches is sampled depending on Sr. Tested with all these ratios (0.2,0.3,0.4,0.5), 0.5 achieved the best results so that was selected for final experiment.

Because sampling is random, subsets may overlap between epochs, but this is not enforced or controlled; the model simply sees a stochastic stream of patches across training.

5.4 Normalization and Train

For unlabeled bd to train I-JEPA mean and std were calculated across all channels for the full set and then mean = (0.28749, 0.31193, 0.23059),std = (0.17089, 0.15828, 0.13506) were used. As it did better on this rather than imagenet mean and std.

Ijepa had been trained for 100 epoch with mostly default optimization settings as official implementation.SGD optimizer with a learning rate of 0.01 momentum 0.09 and 0 weight decay.

5.5 Supervised Training

Segmenter is the main baseline of the project there were two variants created , a vanilla segmenter pretrained with imagenet weights and an IJEPA+Segmenter (I-Seg) pretrained with ijepa but trained on the same exact settings as the vanilla segmenter. The mean and std for both segmenter and I-Seg were the default vit normalization mean = (0.5, 0.5, 0.5), std = (0.5, 0.5, 0.5).

5.6 Hardware and Limitation

Most of the models were trained with a single NVIDIA L40S GPU with 48GB VRAM. The maximum size batch-size for IJepa was 10 and for Segmenter 22. For Most Cases ViT Base was used with patch size 16. The main problem was the limitation in GPUVRAM since these deep learning models uses a lot of VRAM even 48 was not enough. For which batch size were limited which would have shown much more improvement if a greater batch was allowed. However improvement for a ViT base encoder suggests that if a larger encoder like ViT Huge or ViT Giant was used then the mIoU score would be far greater.

6 Result and Analysis

6.1 Loss Curve

6.1.1 Segmenter

Figure 7 shows vanilla Segmenter exhibits a steadily decreasing training loss, but its validation loss gradually rises across epochs, indicating substantial overfitting. This shows that in terms of semantic segmentation of LULC data pretrained Imagenet Weights may not be the right choice for some models.

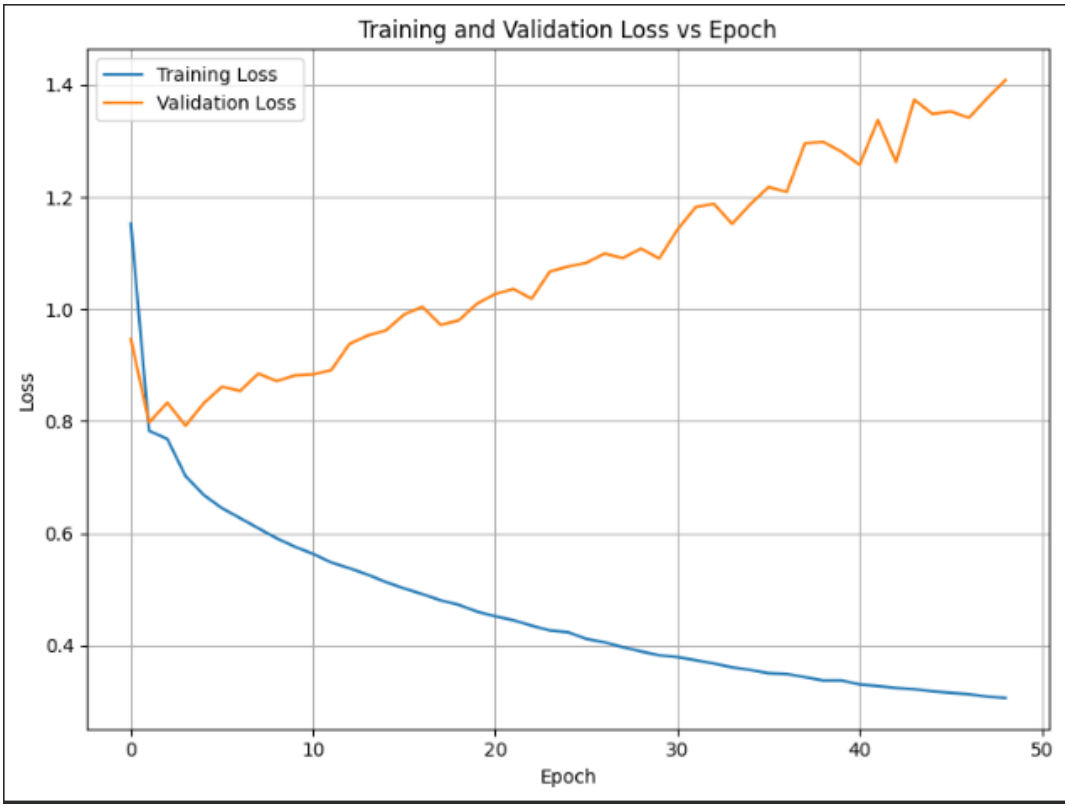


Figure 7: Vanilla Segmenter Loss Curve

6.1.2 Segmenter+Ijepa

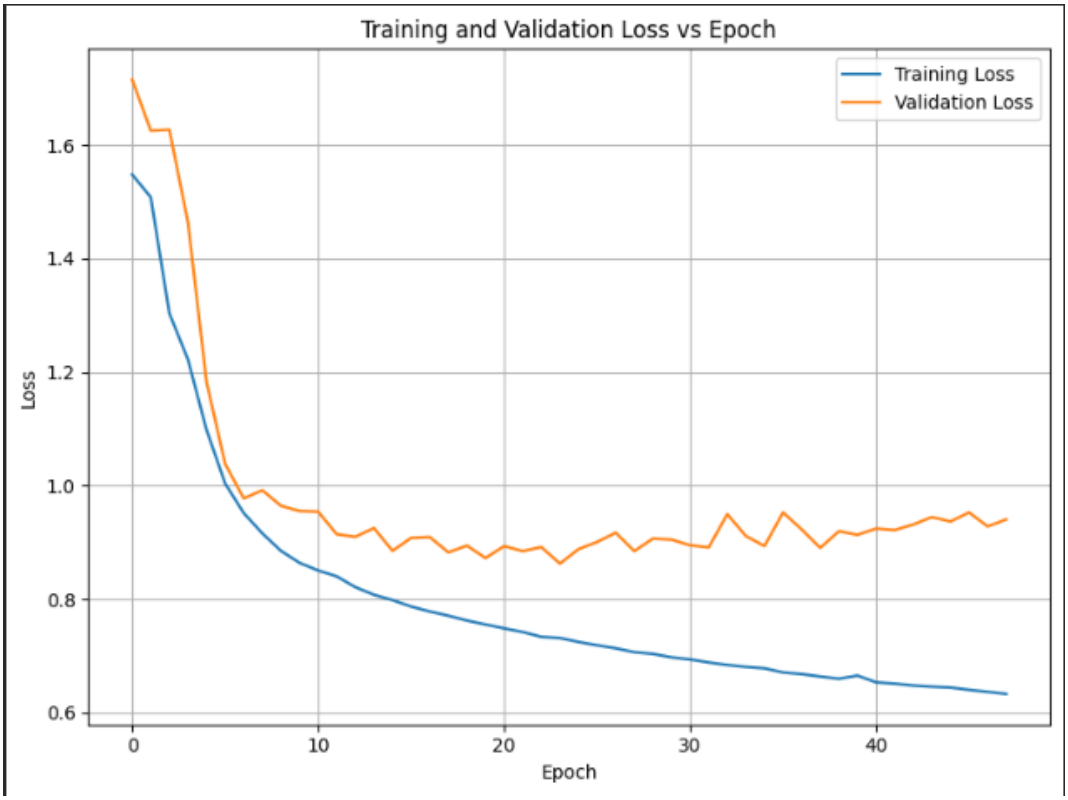


Figure 8: Ijepa Segmenter Loss Curve

Figure 8 shows Segmenter pretrained with I-Jepa, this configuration displays a more stable validation curve, with validation loss decreasing significantly in the early epochs and then oscillating in a narrow range. This suggests that the I-JEPA initialization provides a better starting representation, reducing early overfitting and allowing the model to generalize more consistently throughout training.

Both models were trained for 48 epochs, but the I-JEPA+Segmenter reaches a lower loss plateau substantially earlier, reflecting faster convergence.

6.2 Class-wise Performance

Method	Metric	Forest	Built-up	Water	Farmland	Meadow	Average
Vanilla Segmenter	IoU	0.47	0.56	0.21	0.57	0.23	0.41
	F1	0.64	0.72	0.34	0.72	0.38	0.56
Ijepa + Segmenter	IoU	0.55	0.48	0.30	0.65	0.21	0.44
	F1	0.71	0.65	0.46	0.79	0.35	0.59

Table 2: Per-class IoU and F1 scores for the two models on the BingRGB dataset.

Quantitative results for Classwise IoU and F1 scores are summarized in Table 2. Overall, the I-JEPA-initialized Segmenter outperforms the vanilla Segmenter across nearly all land-cover categories. The improvement is particularly notable in the Forest, Water, and Farmland classes, where the upstream self-supervised pretraining appears to enhance the encoder's ability to capture spatial context and distinguish between structurally similar textures. The vanilla Segmenter achieves a mean IoU of 0.41 and mean F1 score of 0.56, whereas the I-JEPA+Segmenter improves these to 0.44 and 0.59, respectively. Although the absolute gains may appear modest, they are meaningful in remote-sensing segmentation tasks where classes such as water, built-up, and meadow exhibit fine-grained boundaries and intra-class variation. Also this small change to the base encoder shows promise that it will also a lot larger difference in a bigger encoder which has more parameters.

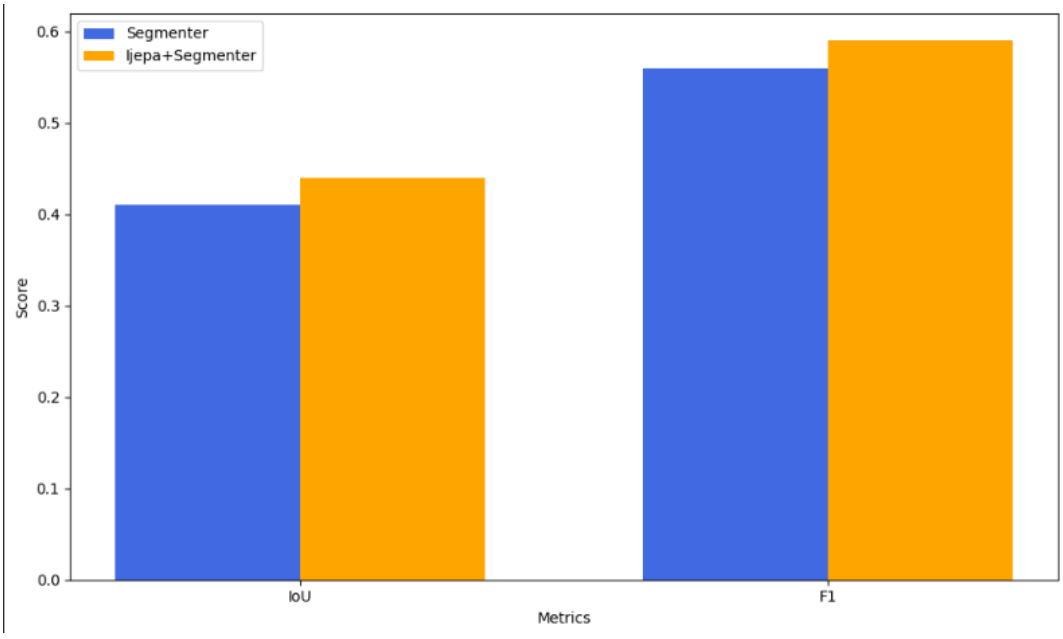


Figure 9: Models BarChart

Figure 9 shows a Barchart shows mean F1 and mean IoU for both the models in the table 2

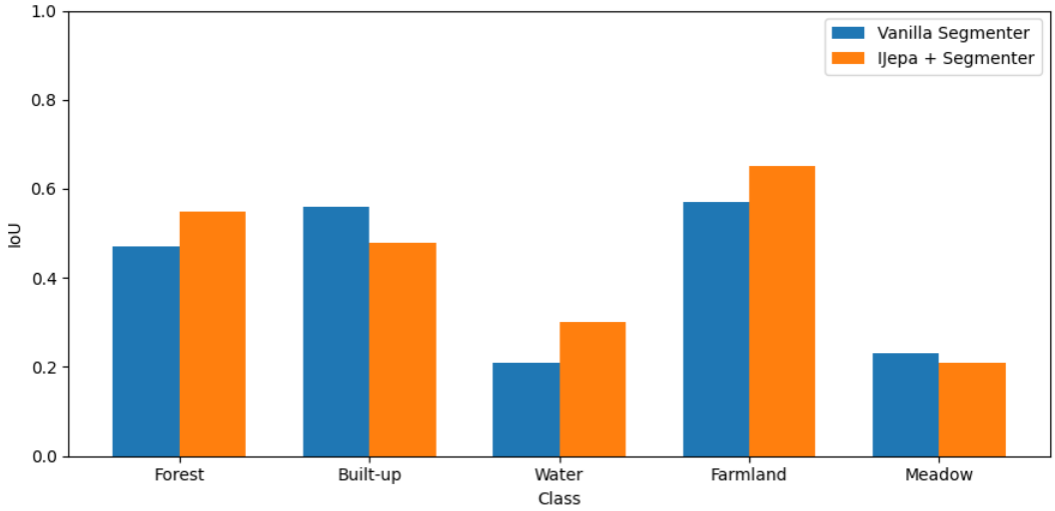


Figure 10: Classwise Barchart

Figure 10 shows the Classwise IoU BarChart of interclasses on segmenter and modified segmenter.

6.3 Predicted Patches

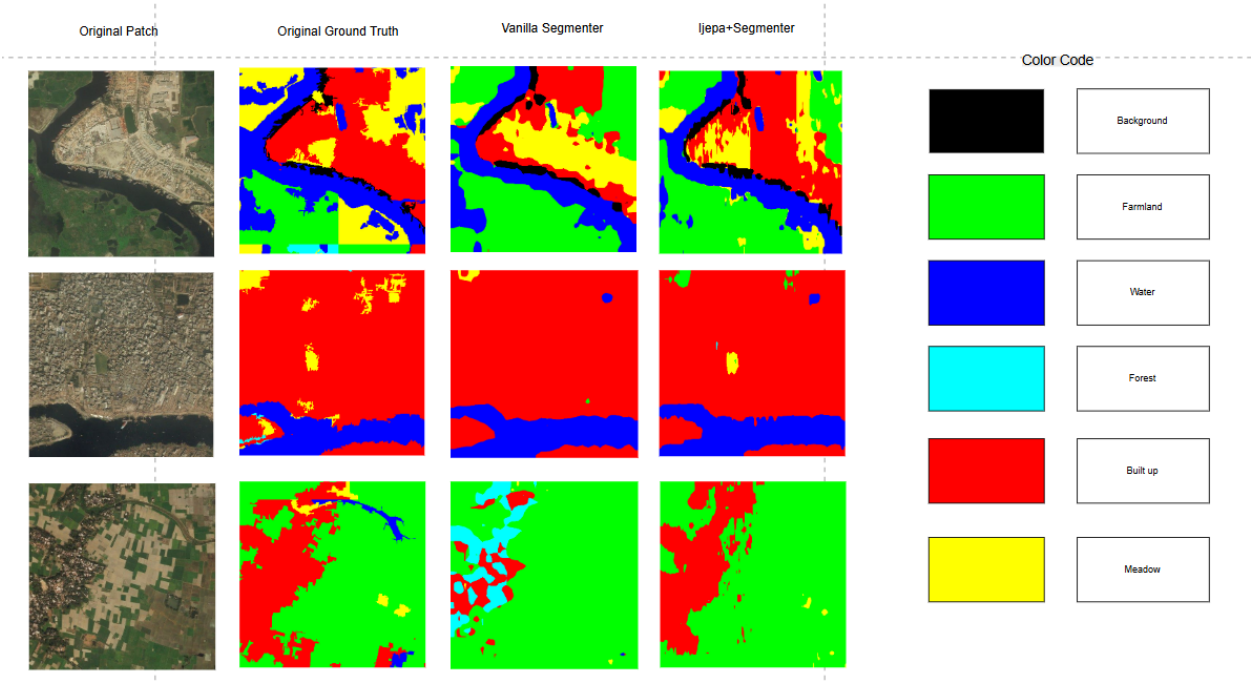


Figure 11: Prediction Patches

6.4 Fail Case

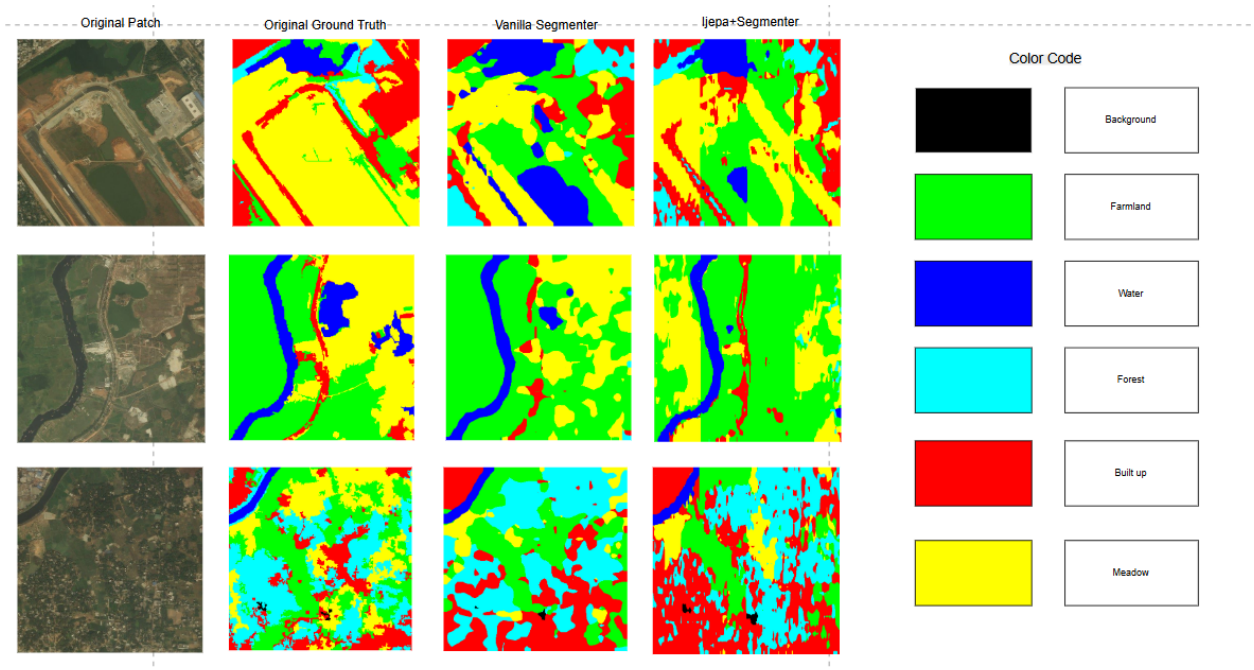


Figure 12: Prediction Patches Fail Case

7 Conclusion and Future Works

7.1 Conclusion

This work investigated whether self-supervised learning can improve transformer-based land-cover segmentation in scenarios where annotated remote-sensing data is limited. By pretraining a Vision Transformer encoder with the I-JEPA framework on a large unlabeled satellite dataset covering the entirety of Bangladesh and subsequently fine-tuning Segmenter on the BingRGB labeled dataset, we demonstrated that self-supervision provides consistent benefits over conventional ImageNet initialization.

Although the gains in mean IoU and F1 score were modest, they were stable across nearly all classes and were accompanied by visibly improved in predicted segmentation maps. These results reinforce the central intuition behind the project: self-supervised pretraining on domain-specific geography enables the model to acquire a contextual understanding of regional land-cover structure before any labeled supervision is applied.

The findings also show that even without extensive hyperparameter tuning the results are relatively promising.

7.2 Future Works

Several directions naturally follow from the outcomes of this study. A straightforward extension is to evaluate I-JEPA initialization on other segmentation frameworks that rely on Vision Transformer backbones.

Many modern architectures such as SegFormer [36], Mask2Former [6], UPerNet-ViT [29], and various hybrid CNN–ViT decoders share the same backbone structure as Segmenter, with some coding we can make them directly compatible with I-JEPA pretrained encoders.

A more ambitious line of work involves replacing I-JEPA with Le-JEPA, the successor framework introduced by the original authors, which has demonstrated stronger representation learning and improved robustness across tasks. Applying Le-JEPA to the same FullBD dataset and fine-tuning Segmenter under identical conditions would yield a clean, controlled comparison of the two paradigms and could potentially amplify the gains seen in this thesis.

References

[1] Assran, M., Duval, Q., Misra, I., Bojanowski, P., Vincent, P., Rabbat, M., LeCun, Y., and Ballas, N. Self-supervised learning from images with a joint-embedding predictive architecture. *arXiv preprint arXiv:2301.08243* (2023).

[2] Carbonneau, P. E., Bergeron, N. E., and Lane, S. N. Texture-based image segmentation applied to the quantification of superficial sand in salmonid river gravels. *Earth Surface Processes and Landforms: The Journal of the British Geomorphological Research Group 30*, 1 (2005), 121–127.

[3] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision* (2021), pp. 9650–9660.

[4] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)* (September 2018).

[5] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (2020), PmLR, pp. 1597–1607.

[6] Cheng, B., Misra, I., Schwing, A. G., Kirillov, A., and Girdhar, R. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2022), pp. 1290–1299.

[7] Cong, Y., Khanna, S., Meng, C., Liu, P., Rozi, E., He, Y., Burke, M., Lobell, D., and Ermon, S. Satmae: Pre-training transformers for temporal and

multi-spectral satellite imagery. *Advances in Neural Information Processing Systems 35* (2022), 197–211.

[8] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 3213–3223.

[9] Demir, I., Koperski, K., Lindenbaum, D., Pang, G., Huang, J., Basu, S., Hughes, F., Tuia, D., and Raskar, R. Deepglobe 2018: A challenge to parse the earth through satellite images. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (2018), pp. 172–181.

[10] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (2009), Ieee, pp. 248–255.

[11] Doma, V., Berbić, E., Abd Almisreb, A., Saleh, M. A., and Amanzholova, S. A review of ai transformers in image segmentation. *neural networks 6* (2023), 7.

[12] Dosovitskiy, A. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).

[13] Graser, A., Sutton, T., and Bernasocchi, M. The qgis project: Spatial without compromise. *Patterns 6*, 7 (2025), 101265.

[14] Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems 33* (2020), 21271–21284.

[15] He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2022), pp. 16000–16009.

[16] He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 9729–9738.

[17] Helber, P., Bischke, B., Dengel, A., and Borth, D. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 12*, 7 (2019), 2217–2226.

[18] Hong, D., Zhang, B., Li, H., Li, Y., Yao, J., Li, C., Werner, M., Chanussot, J., Zipf, A., and Zhu, X. X. Cross-city matters: A multimodal remote sensing benchmark dataset for cross-city semantic segmentation using high-resolution domain adaptation networks. *Remote Sensing of Environment 299* (2023), 113856.

[19] Kingma, D. P., and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[20] Liu, F., Chen, D., Guan, Z., Zhou, X., Zhu, J., Ye, Q., Fu, L., and Zhou, J. Remoteclip: A vision language foundation model for remote sensing. *IEEE Transactions on Geoscience and Remote Sensing 62* (2024), 1–16.

[21] Liu, S., Chen, L., Zhang, L., Hu, J., and Fu, Y. A large-scale climate-aware satellite image dataset for domain adaptive land-cover semantic segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing 205* (2023), 98–114.

[22] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision* (2021), pp. 10012–10022.

[23] Long, J., Shelhamer, E., and Darrell, T. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015), pp. 3431–3440.

[24] Manas, O., Lacoste, A., Giró-i Nieto, X., Vazquez, D., and Rodriguez, P. Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 9414–9423.

[25] Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[26] Paul, O., Nayem, A. B. S., Sarker, A., Rajib, R. H., Jit, T. H., Alam, A., Amin, M. A., Ali, A. A., and Rahman, A. M. Bolm high resolution land use and land cover dataset and benchmark results for the rapidly developing city of dhaka bangladesh. *Scientific Reports 15* (2025), 37689.

[27] Regmi, K., and Shah, M. Bridging the domain gap for ground-to-aerial image matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 470–479.

[28] Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)* (2015), Springer, pp. 234–241.

[29] Ruiping, Y., Kun, L., Shaohua, X., Jian, Y., and Zhen, Z. Vit-upernet: a hybrid vision transformer with unified-perceptual-parsing network for medical image segmentation. *Complex & Intelligent Systems 10*, 3 (2024), 3819–3831.

[30] Strudel, R., Garcia, R., Laptev, I., and Schmid, C. Segmenter: Transformer for semantic segmentation. *arXiv preprint arXiv:2105.05633* (2021).

[31] Tarvainen, A., and Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems 30* (2017).

[32] Van Etten, A., Lindenbaum, D., and Bacastow, T. M. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232* (2018).

[33] Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence 43*, 10 (2020), 3349–3364.

[34] Wang, J., Zheng, Z., Ma, A., Lu, X., and Zhong, Y. Loveda: A remote sensing land-cover dataset for domain adaptive semantic segmentation. *arXiv preprint arXiv:2110.08733* (2021).

[35] Xia, J., Yokoya, N., Adriano, B., and Broni-Bediako, C. Openearthmap: A benchmark dataset for global high-resolution land cover mapping. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (2023), pp. 6254–6264.

[36] Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., and Luo, P. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems 34* (2021), 12077–12090.

[37] Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L. M., and Shum, H.-Y. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605* (2022).

[38] Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., and Kong, T. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832* (2021).