Machine Learning Project Report

Nguyen Duc Anh Pham 7059450 Quoc Viet Dao 7010834
Khanh Huyen Vo Thuc

Summer 2025

Abstract

In the field of computer vision, capturing the world as humans perceive and understand it has consistently been cornerstone of groundbreaking advancements. One of the pivotal techniques that enable this understanding is *image segmentation*.

1 Introduction

Image segmentation is a fundamental task in computer vision, aimed at partitioning an image into meaningful regions for analysis and interpretation. Traditional segmentation methods often require either fully annotated datasets or rely on unsupervised approaches, both of which have limitations in scalability and accuracy. This occurs on a pixel level to define the precise outline of an object within its frame and class. Those outlines, otherwise known as the output, are highlighted with either one or more colors, depending on the type of segmentation.

In this experiment, we aim to explore scribble-based image segmentation, a semi-supervised technique where sparse annotations (scribbles) guide the segmentation process using **UNet** [1], **ResUNet** [2] and **Attention ResUNet**[3]. Our pipeline starts with the step of data augmentation, where we use techniques like Geometric Transformations, Color Space Augmentation, Noise Injection, Blurring/Sharpening, and Erasing to create modified versions of the original images from the dataset. We also use the technique of Self-Supervised Learning (SSL) **distillation with no labels (DINO) [4]** to give the model a strong prior about the visual features of the data: edges, textures, shapes without manual labeling. After the model is first trained using self-supervised learning to extract rich features, it is then fine-tuned on images and scribbles for our downstream task of Image Segmentation with Scribble.

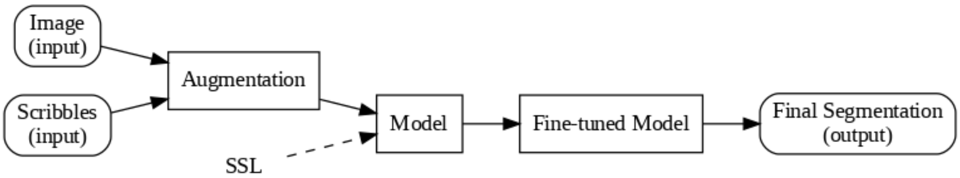


Figure 1: Pipeline

2 Data Processing

The dataset under analysis consists of 228 *training samples* and 226 *test-1 samples*, each containing an *RGB* image, sparse scribble annotations, and ground truth segmentation masks. The consistency in image dimensions is remarkable, where all samples maintain a uniform size of 500×375 pixels, indicating a carefully curated dataset designed for consistent processing

pipelines. Without any further preprocessings, we directly use augmentations techniques to create variants of the original images and scribbles.
Our Data Augmentation techniques are the following:

- ResizeRandomCrop: Crop a random portion of image and resize it to a given size.
- HorizontalFlip: Horizontally flip the given image randomly with a given probability.
- VerticalFlip: Vertically flip the given image randomly with a given probability.
- ColorJitter: Randomly change the brightness, contrast, saturation and hue of an image.
- RandomGrayscale: Randomly convert image to grayscale with a probability of p (by default 0.1).
- GuassianBlur: Blurs image with randomly chosen Gaussian function for blurring.

3 Methodology

3.1 Model

We use *Pytorch* to implement UNet and two of its variants ResUNet and AttentionResUNet to provide stronger feature extraction and the latter with attention mechanisms, improving performance on complex datasets. We use these models for this image segmentation task because these architectures are specifically designed to capture both global context and fine-grained spatial details, enabling them to propagate sparse scribble annotations into accurate, full-resolution pixel-wise masks efficiently and robustly.

3.1.1 UNet

U-Net is a convolutional neural network designed for pixel-level prediction. It has a U-shaped architecture: an encoder compresses or downsamples the input into high-level features, and a decoder upsamples those features back to the original resolution, with skip connections that pass fine details from encoder to decoder.

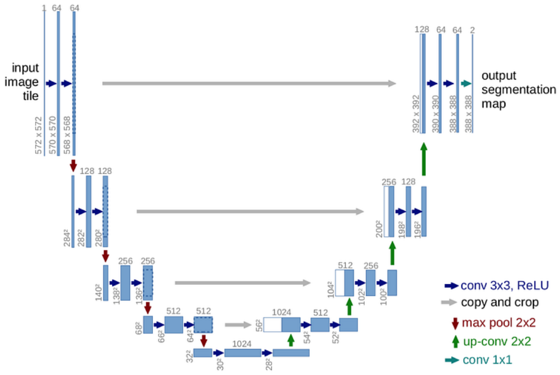


Figure 2: UNet[1]

3.1.2 Residual UNet

It is a variation of UNet that incorporates *residual connections*[5] within the architecture. These residual connections can help to alleviate the vanishing gradient problem and improve the overall performance of the original UNet.

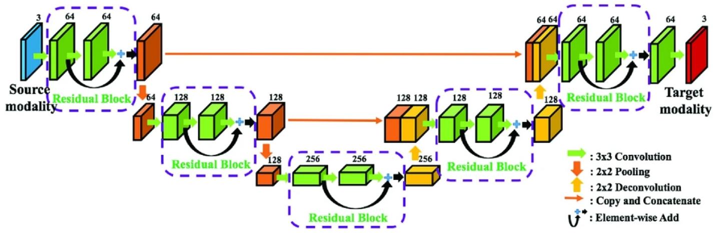


Figure 3: Residual UNet[2]

3.1.3 Attention Residual UNet

It is a combination of ResUNet and Attention UNet, it incorporates both *residual connections* and *attention mechanisms*[6]. It can explicitly guide the learning process of each decoder layer. The individual loss function to each decoder layer helps to supervise the learning process of each layer in the decoder and thereby enables them to generate better feature maps. The attention gates in the generator focuses on the activation of relevant information instead of allowing all information to pass through the skip connections in the ResUNet.

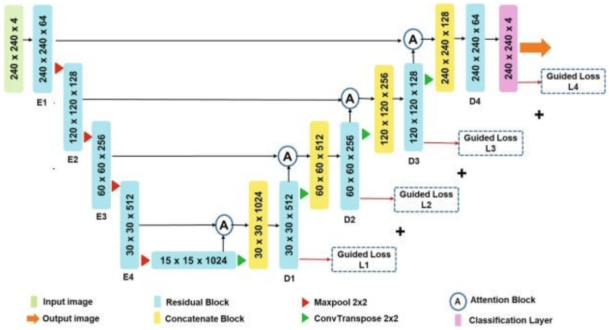


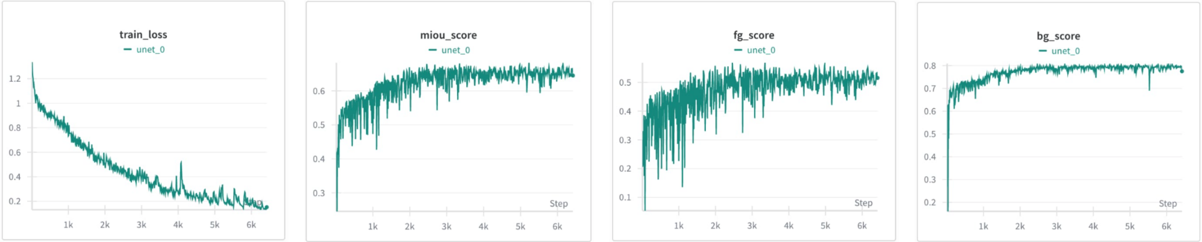
Figure 4: Attention Residual UNet[3]

3.2 Self-supervised Learning with DINO

DINO's pipeline[4] divides into two parts, a student network and a teacher network. Both networks are vision transformers [7], which are designed to process images by treating them as sequences of patches. The model passes two different random transformations of an input image to the student and teacher networks. Both networks have the same architecture but different parameters. The output of the teacher network is centered with a mean computed over the batch. Each networks outputs a K dimensional feature that is normalized with a temperature softmax over the feature dimension. Their similarity is then measured with a cross-entropy loss. We apply a stop-gradient (sg) operator on the teacher to propagate gradients only through the student. This collaborative setup allows the model to learn robust and invariant features from the data without the need for manual labels, thereby achieving effective self-supervised learning.

3.3 Training

We trained three models using the *DiceBCE loss* [8], which combines the Dice–Sørensen coefficient with Binary Cross-Entropy loss. All models were optimized with *AdamW*[9]. Our training strategy consisted of two stages: first, we performed self-supervised pretraining by training DINO for 100 epochs, using the resulting weights as initialization for the downstream segmentation task. In the subsequent full fine-tuning stage, we trained the models with strong data augmentation. Although this slowed early convergence, performance steadily improved with ex-



(a) Train loss (b) Mean IoU (c) ForeGround IoU (d) BackGround IoU

Figure 5: Observing the training process with the U-Net model.

tended training. Given our computational resources, we opted to train for 1,000 epochs during supervised learning.

Observing the training process reveals the impact of strong augmentation, where frequent transformations of the foreground objects lead to significant fluctuations and instability in the foreground IoU after each update step. In contrast, the model shows more consistent improvement in recognizing background regions.

4 Evaluation

Using the provided **mean Intersection over Union (IoU)** of Background and Object. The evaluation results in these two tables are based on comparing the model outputs with manually annotated ground-truth labels.

Without SSL			
	Object IoU	**BackGround IoU**	**mIoU**
UNet	0.53	0.828	0.679
ResUNet	0.562	0.684	0.623
AttnResUNet	0.575	0.743	0.659

Table 1: Results without SSL infused

Our results suggest that, with limited training data and heavy augmentation, DINO tends to degrade background IoU, resulting in a noticeable drop in overall performance.

With SSL			
	Object IoU	**BackGround IoU**	**mIoU**
UNet	0.492	0.63	0.561
ResUNet	0.591	0.575	0.583
AttnResUNet	0.631	0.613	0.622

Table 2: Results with SSL infused

5 Conclusion and outlook

In this challenge, we explored data augmentation and self-supervised learning (SSL) to address limited and low-diversity data. However, our results indicate that with such a small dataset, SSL-pretrained backbones did not achieve sufficient generalization to benefit downstream tasks. Moreover, each approach influences the model in different ways. By observing these effects, we gain valuable insights that can inform and improve future projects.

References

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[2] Lei Xiang, Yang Li, Weili Lin, Qian Wang, and Dinggang Shen. Unpaired deep cross-modality synthesis with fast training. In *International Workshop on Deep Learning in Medical Image Analysis*, pages 155–164. Springer, 2018.

[3] Dhiraj Maji, Prarthana Sigedar, and Munendra Singh. Attention res-unet with guided decoder for semantic segmentation of brain tumors. *Biomedical Signal Processing and Control*, 71:103077, 2022.

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[8] Carole H. Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M. Jorge Cardoso. *Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations*, page 240–248. Springer International Publishing, 2017.

[9] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.