Technical Assessment

for Meritiinc

Dr. Mussadiq Abdul Rahim

February 13, 2025

---

1 Problem Statement

This problem is a segmentation problem for which we need to label each pixel. Unlike traditional methods in deep learning which require full segmentation masks as anchors for model training, this task considers incomplete segmentations, using points for annotation. Such a situation is rarely seen in deep learning and poses unique challenges because the information provided by the point labels is limited.

The task detail mentions a intended solution for a project referred to as LandVisor project. It must utilize transfer learning with pre-trained models on sufficiently large datasets (e.g. example ImageNet) in order to ensure a proper baseline. A semi-supervised learning methods is suggested to be used to label unmarked pixels. An additional loss function is given to be included in the actual loss. Partial Cross Entropy (CE) Loss is defined as follows.

$$\text{pfCE} = \frac{\sum (\text{FocalLoss}(\text{pre}, \text{GT}) \cdot \text{MASK}_{\text{labeled}})}{\sum \text{MASK}_{\text{labeled}}}$$

Where pre is prediction, GT is ground truth, $\text{MASK}_{\text{labeled}}$ is the mask for an image.

In addition, to this loss function following parts of project are indicated.

- Use amplification when testing to improve performance.
- Use ensemble learning to integrate multiple deep learning models to further improve performance.
- Referring to the original ML pipeline, histogram matching is used to ensure the stability of the picture style.

However, the **Task** indicates implementation as follows:

- Implement the partial Cross Entropy loss.
- Find any remote sensing image segmentation data.
- Randomly sample the simulated point labels.
- Add the loss to any remote sensing segmentation network.
- Design experiments and explore one or two factors that affect the performance.
- Write technical report to include:
  - Method
  - Experiment (purpose/hypothesis, experimental process, results)

2 Tools and Technologies

As guided that only Python is allowed for use to solve this technical assessment. Whereas any deep learning libraries could be adapted for this purpose. Following tools and technologies were used.

- **segmentation-models** [1] for pre-trained models.
- **tensorflow** [2] tensorflow for deep learning backend.
- **keras** [3] for deep learning.

3 Deliverables

In fulfillment of the task

- Implement the partial Cross Entropy loss. Add the loss to any remote sensing segmentation network.

```
def semi_supervised_loss(y_true, y_pred, mask_labeled,
    alpha=0.25, gamma=2.0):
```

Listing 1: "See code"

- Find any remote sensing image segmentation data. Resized version of AIRS is [4] selected.

  - Publicly available at kaggle.com.
  - Annotated.
  - Aerial images with over 220,000 buildings.
  - Easily mountable to Google Drive and Colab (free plan).

- Randomly sample the simulated point labels. Design experiments and explore one or two factors that affect the performance. See section 4.

- Write technical report: This document.

4 Experiments

4.1 Experiment A

Following factors under study in this experiment.

```
EPOCHS = 25
ALPHA = 0.25
GAMMA = 2.0
ALPHA_TEACHER = 0.99
optimizer = tf.keras.optimizers.Adam(learning_rate=1e-3)
```

The model shows consistent improvement over the 25 epochs, with both supervised and consistency losses steadily decreasing, indicating effective learning and generalization. The increase in consistency loss suggests that the model is becoming more robust to the input variations as training progresses.

4.2 Experiment B

```
EPOCHS = 25
ALPHA = 0.1
GAMMA = 2
ALPHA_TEACHER = 0.99
optimizer = tf.keras.optimizers.Adam(learning_rate=1e-3)
```

Experiment B shows faster initial reduction in total loss and consistency loss compared to Experiment A, likely due to the smaller ALPHA value (0.1 vs. 0.25). This could indicate a more rapid adaptation towards consistency, while Experiment A focuses more on supervised learning. Experiment B achieves a lower final total loss (0.1322) than Experiment A (0.3450), though both experiments exhibit a strong trend of decreasing supervised and consistency losses over epochs.

4.3 Experiment C

```
EPOCHS = 25
ALPHA = 0.25
GAMMA = 8
ALPHA_TEACHER = 0.99
optimizer = tf.keras.optimizers.Adam(learning_rate=1e-3)
```

Experiment C shows a notable improvement in both supervised and consistency losses, achieving a lower total loss (0.0254) compared to Experiment A (0.3450) and Experiment B (0.1322). The model in Experiment C, with a higher GAMMA (8), leads to a sharper decrease in consistency loss, suggesting a more aggressive push for robust learning and generalization. Supervised loss also decreases consistently, demonstrating effective supervised learning in combination with high consistency. The model in Experiment C is more efficient in reducing both types of losses within fewer epochs.

4.4 Experiment D

```
EPOCHS = 25
ALPHA = 0.1
GAMMA = 8
ALPHA_TEACHER = 0.99
optimizer = tf.keras.optimizers.Adam(learning_rate=1e-3)
```

Experiment D demonstrates exceptional performance with the lowest total loss (0.0034) compared to Experiments A, B, and C. Despite using a smaller ALPHA (0.1), the higher GAMMA (8) significantly reduces consistency loss at an impressive rate, which is evident in the sharp decline in both supervised and consistency losses throughout the epochs. The combination of a smaller ALPHA and higher GAMMA in Experiment D results in a more refined balance, achieving the lowest total loss and demonstrating the most efficient trade-off between supervised learning and consistency regularization across all experiments.

4.5 Experiment E

```
EPOCHS = 25
ALPHA = 0.5
GAMMA = 0.5
ALPHA_TEACHER = 0.99
optimizer = tf.keras.optimizers.Adam(learning_rate=1e-3)
```

Experiment E achieves the lowest total loss across all experiments, with a final total loss of 0.0005. This experiment combines a higher ALPHA (0.5) and a lower GAMMA (0.5), which appears to provide an excellent balance between supervised and consistency losses. The model consistently improves both losses throughout the epochs, indicating efficient regularization and stable learning. Compared to Experiments A, B, C, and D, Experiment E stands out for achieving a significant reduction in both supervised and consistency losses, particularly after the first few epochs. It has the lowest supervised loss (0.0003), alongside a notable decline in consistency loss (from 0.0071 to 0.0015), showcasing exceptional performance in optimizing both components without sacrificing accuracy.

References

[1] P. Iakubovskii, "Segmentation models," 2019.

[2] "TensorFlow: Large-scale machine learning on heterogeneous systems."

[3] F. Chollet *et al.*, "Keras," 2015.

[4] Q. Chen, L. Wang, Y. Wu, G. Wu, Z. Guo, and S. L. Waslander, "Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 147, pp. 42–55, 2019.