

## Assignment 11: Build UNet-ID Instance Segmentation Architecture

40 Points

**Deliverable:** Notebook (.ipynb file) with all required code to complete the stated tasks. Answer all questions in Markdown cells.

**Overview:** The UNet-Id architecture augments UNet to support instance segmentation as opposed to semantic segmentation. The goal of this exercise is to build this architecture from scratch by subclassing the `nn.Module` class.

The architecture was introduced in the following paper:

Wagner, F.H., Dalagnol, R., Tarabalka, Y., Segantine, T.Y., Thomé, R. and Hirye, M.C., 2020. U-net-id, an instance segmentation model for building extraction from satellite images—case study in the joanópolis city, brazil. *Remote Sensing*, 12(10), p.1544.

The paper is available in open access: <https://www.mdpi.com/2072-4292/12/10/1544>.

### Background Questions

**B1:** What is the difference between semantic and instance segmentation? (2.5 Points)

**B2:** Write a paragraph explaining the components of the traditional UNet architecture. Please discuss the architecture in detail. Someone should be able to use your description to build the architecture by subclassing `nn.Module`. (5 Points)

**B3:** Using the paper cited above, explain how UNet-ID augments the UNet architecture for instance segmentation. Please discuss the architecture in detail. Someone should be able to use your description to build the architecture by subclassing `nn.Module`. (5 Points)

**B4:** Explain the input data requirements for UNet-ID and how the requirements are different than UNet. (2.5 Points)

### Task

Build the UNet-ID architecture from scratch by subclassing `nn.Module`. Make sure to include comments throughout to explain the architecture. **Up to 25 points will be awarded** based on the correctness of your implementation and how well it is documented/explained with comments. You should test the architecture by instantiating it and predicting some randomly generated data of the correct shape.

### Hints and Suggestions

1. Try drawing out the architecture on paper before building it out in code.
2. It is easiest to build the architecture up in pieces. You do not have to build the entire architecture in a single `nn.Module` subclass.
3. Make sure to reuse components when possible.

4. Figures 1 and 2 in the paper and the associated text should be your primary source material.
5. The model should generate three output predictions.