This tutorial is designed as a primer on how to use ArcGIS Pro Deep Learning Tools to create an object detection model. These tools are designed to automate digitization such as extracting building footprints from aerial imagery. These tools are based on specific software extensions and hardware. ESRI ArcPro is the base GIS software and the AI/Deep Learning extensions are in the Deep Learning Libraries (PyTorch and Tensor Flow may be familiar) also required is a Nvidia Graphics Card (with Cuda Cores and minimum 12GB VRAM) to run AI training models locally. As with most analysis tools the more RAM, VRAM and CPU power the better. A good understanding of GIS remote sensing analysis, ArcPro and excelent file management skills are required for this tutorial. The deep learning libraries can also be run remotely on cloud-based services, but that functionality is beyond the scope of this document.

Esri Links that offer further explanation of DL tools and the libraries to download:

- Intro to deep learning
- Deep learning in ArcGIS Pro
- Deep learning libraries installer
- Building object classification model tutorial video
- Building Footprints Extraction PDF Tutorial

The following will go into basic detail on how to use the ArcGIS Pro Deep Learning Tools to create an object detection model. There are 3 main steps:

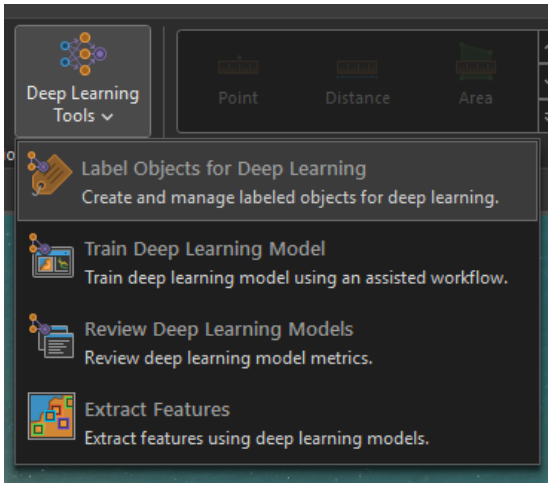
1. Collect training samples
2. Create model
3. Test model

The first step of collecting training samples can be done on any computer with ArcGIS Pro installed, however, the following 2 steps require the Deep Learning Libraries to be installed with the link found above. This library requires a computer with a Nvidia graphics card to function.

Collecting Training Samples

The following are the basic steps for collecting training samples to be used to train a deep learning model. These samples determine your model's ability to accurately detect objects, and as such, it is essential that all samples created by you are as accurate as possible. Generally, the more accurate samples that are provided when creating the model the better the model will be.

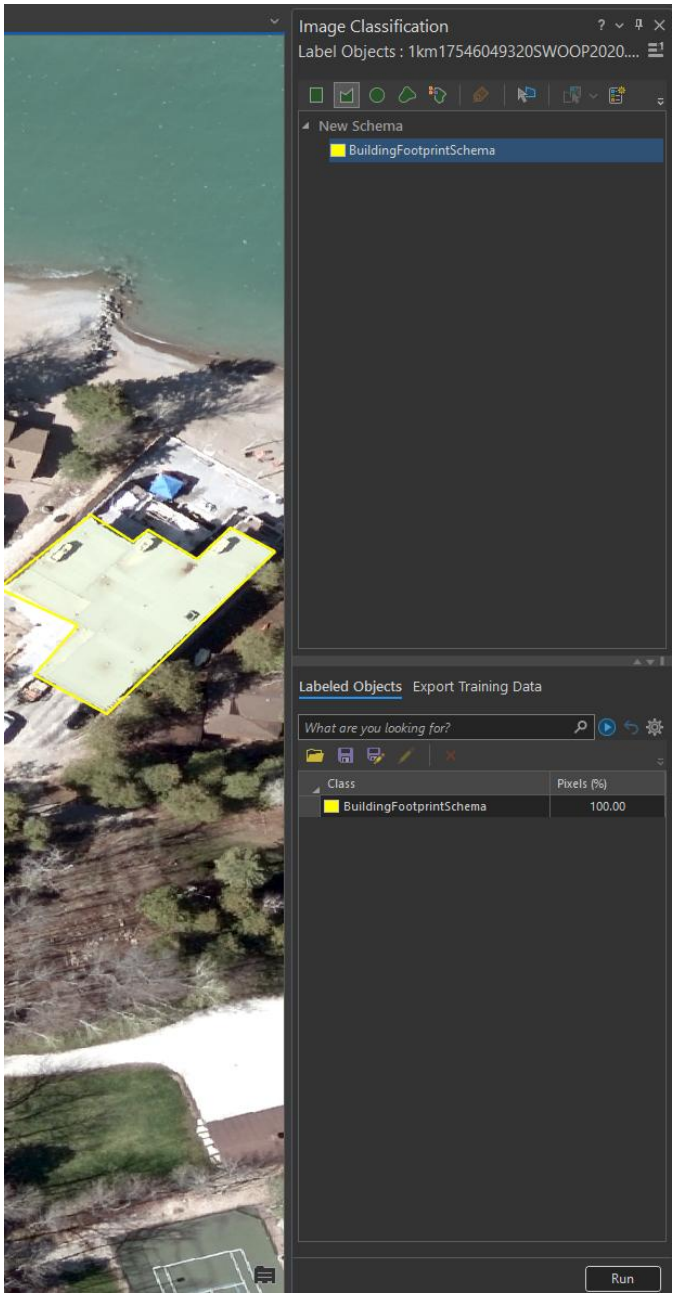
1. Open ArcGIS Pro
2. Add the image from which you will be collecting your samples via the catalog pane. Preferably, an image that has many buildings that aren't obscured by vegetation just to give your first attempt a better chance.
3. Navigate to imagery in the top bar and click on "Deep Learning Tools" -> "Label Objects for Deep Learning"
4. In the pop-up window leave the "Label using" field as the default of "Existing imagery layer" and select the image which you added earlier in the "Imagery layer" field. Click "OK"



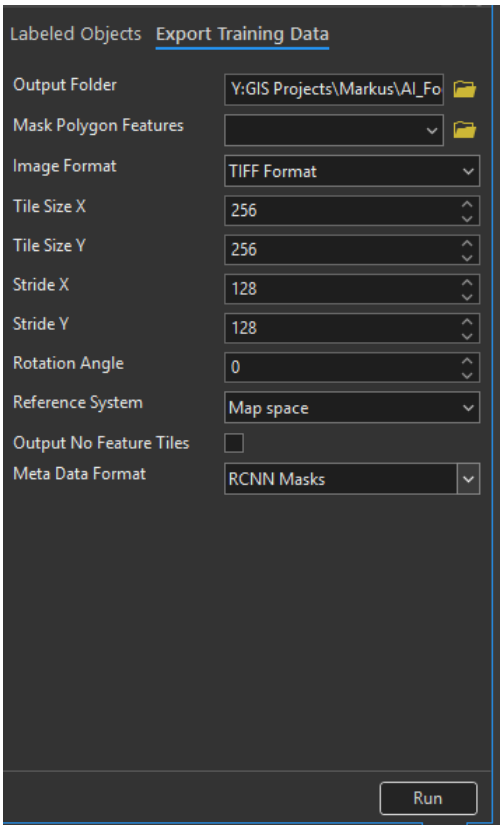
5. This will open the "Image Classification" pane with the selected image as the image being classified. This pane has many components to it which will be covered briefly below:
    a. Create a .gdb in a folder within your working folders to save the Image Classification Schema
    b. At the top will be the name of the image being classified.
    c. Below are multiple selection tools which are used to create boundaries around samples.
    d. Below is the schema panel, which will be covered in more detail later.
    e. Below is the "Labeled Objects" pane where all created samples will appear and can be edited, deleted or saved.
    f. Finally, is the "Export Training Data" pane where the collected samples will be exported to be used in step 2.
6. Before starting to collect samples, a new schema must be created. A schema is the category in which the samples being collected will be stored. Multiple schemas can be created if there are different types of sample categories for different types of

classifications. However, this tutorial is focusing on purely object detection rather than classification, so only 1 schema will be necessary.

7. To create the schema, right click on “New Schema” and click “Add New Class”. Choose a name and colour for the category and set the “Value” field to 1. Leave all other fields blank. Click “OK”
8. Now, sample collection can begin. Ensure that the schema that was created is selected in the schema pane and choose a selection tool most appropriate for the sample being collected. As mentioned earlier, the more accurate the samples are the better the model will be, so, choosing something like the “Polygon” selection tool and getting very accurate sample boundaries will improve the model. Although more accurate samples will take longer to collect.
9. During the sample collection process, it is a good idea to save often using the save button under the “Labeled Objects” pane. These saved samples can be opened again using the “Load Training Samples” button under the same pane if sample collection takes place across multiple ArcGIS Pro sessions. Samples are saved separately from the project in ArcGIS Pro so you must ensure that both are saved before closing the project or else the samples will be lost.



10. Once the necessary samples are collected, they can be exported. Save the samples once again and click “Export Training Data”. Create and choose an “Output Folder” for the samples to be saved into, this is a separate save then just saving the samples as this exports the samples to be used in other Deep Learning Tools. All fields until “Meta Data Format” can be left as default. For this tutorial, “Meta Data Format” will be set as “RCNN Masks”. This meta data format is often best for object detection, information about meta data formats can be found in the Export Training Data tool’s documentation. Click “Run”.



Training samples have now been collected and exported to be used in training a model in the next step.

RCNN Mask: https://developers.arcgis.com/python/latest/guide/how-maskrcnn-works/

Training a Deep Learning Model

The following are the basic steps for training a deep learning model using the geoprocessing tool with the same name in ArcGIS Pro. This step requires the deep learning libraries installed and a Nvidia graphics card.

1. Click “Analysis” in the top bar and select “Tools”
2. Search for “train deep” and select the top result of “Train Deep Learning Model”
3. Under the “Input Training Data” field select the exported training samples from the last step. Multiple training samples can be selected to improve the model.
4. After the input data is selected, ArcGIS Pro will automatically add new settings in the tool to match the meta data format that the samples were exported with. Leave these settings as default.
5. Create and choose an “Output Folder” for where the final model will be stored

6. Ensure that “Model Type” has been automatically selected as “MaskRCNN (Object detection)”
7. All other fields can be left as default. When creating a model outside of this tutorial changing fields such as “Max Epochs” or “Batch Size” can improve the model. More information on the other field can be found under the help documentation of the tool by clicking the question mark icon in the top right corner.
8. Click “Run”. Depending on how many samples are provided this tool can take multiple hours to run. Once created, the model can be tested.

Using a Deep Learning Model

The following are the basic steps for how to use a Deep Learning Model to detect objects in images.

1. Click “Analysis” in the top bar and select “Tools”
2. Search for “detect objects” and select the top result of “Detect Objects Using Deep Learning”
3. Under the “Input Raster” field, select the image from which you want to detect objects in.
4. Under the “Output Detected Objects” field, choose the name and location for where you want the detected objects to be stored. Detected objects will be stored in a polygon shapefile.
5. Under the “Model Definition” field, navigate to the folder where you stored the created model from the previous step and select the “.dlpk” file in that folder. A “.dlpk” file extension is the extension which ArcGIS Pro uses for deep learning models.
6. All other fields can be left as default.
7. Click “Run”.

The output of this tool will be the objects that were detected using your deep learning model. The number of training samples along with the settings of the “Training a Deep Learning Model” tool will determine the accuracy and precision of the detected objects.

Results

Below is the resultant output the “Detect Objects Using Deep Learning” using the training data created in the first step of this primer. The red outlined polygons are the digitized sample footprints used to train the AI on the various different roof colours. The yellow polygons are the trained AI output. The results will need refinement possibly using the “Simplify Polygon”

tool to straighten lines and the digitization of the missing footprints. Also, refining the training model by adding samples that stand out as missed by the AI Detection. This is an iterative process where we train ourselves on how to train this Deep Learning AI to eventually automate extracting building footprints from imagery.

This primer covered a typical scenario for the using the Deep Learning tools within the ArcPro suite. If you have any questions on learning more about ArcGIS and how to incorporate it into research, please contact us.

Contact Us
Geospatial Centre
Dana Porter Library, Room 328
University of Waterloo Library
Waterloo, Ontario N2L 3G1
LibraryGEO@uwaterloo.ca

Created by Batu Ozer and Markus Wieland for the University of Waterloo Geospatial Centre