**How to write custom python raster function and esri model definition file**

**Python Raster Function(inference function)**

Look into DraftTemplateObjectDetector.py. Needs update later.

**Esri Model Definition**

To perform inferencing in ArcGIS Pro, the Detect Object for Deep Learning tool requires an esri model Definition (.EMD) file. Download Template emd file from <https://github.com/Esri/deep-learning-frameworks> and make the following changes

Required:

* **"ModelFile"**: Type: String. **ModelFile** is the path to a trained deep learning model file or folder(depends on what deep learning framework you are using and whether it requires loading a model folder or a model file) name. The file format depends on the model framework. For example, in TensorFlow, the model file is a .pb file. If it shares the same parent folder with emd file, "ModelFile": ".\\frozen\_inference\_graph.pb" or just "ModelFile": "frozen\_inference\_graph.pb". If it is in another location, please use absolute path: "ModelFile": "D:\\test\\frozen\_inference\_graph.pb".
* **"InferenceFunction"**: Type: String. **InferenceFunction** is the path of the custom Python Raster Function(inference function). An inference function understands the trained model data file and provides the inferencing logic. If it shares the same parent folder with emd file, " InferenceFunction": ".\\ArcGISObjectDetector.py" or just " InferenceFunction": "ArcGISObjectDetector.py". If it is in another location, please use absolute path: "InferenceFunction": "D:\\test\\ArcGISObjectDetector.py".
* **“ModelType”**: Type: String. **ModelType** is the type of model. For detecting objects or features, “ModelType”: “ObjectDetection”. For classifying pixels , “ModelType”: “ImageClassification”. For classifying objects and features, “ModelType”: “ObjectClassification”.
* **“ImageHeight”**: Type: Int. **ImageHeight** is the number of rows in the image that the trained model is trained on. For example, 512.
* **“ImageWidth”**: Type: Int. **ImageWidth** is the number of columns in the image that the trained model was trained on. For example, 512.
* **“Classes”**: Type: List of tuples. **Classes** contains the information about the output class categories or objects. For each class, "Name" uses the same label name you used for training, "Value" should always be 1, 2, 3, 4, etc..., and “Color” can be any RGB combination.

For example:

"Classes": [

{

"Value": 1,

"Name": "Airplane",

"Color": [0, 255, 0]

},

{

"Value": 2,

"Name": "Calf",

"Color": [255, 255, 0]

}

]

Optional:

* "**Padding**": Type: Int. Default: 0. **Padding** is the number of extra pixels needed on each side of input pixel blocks. The amount of padding to add to the input imagery for inferencing. For example, if a model outputs the center 128x128 segmentation pixels given an input tile 256x256, the model has a padding of 64.
* "**Threshold**": Type: Float. Default: 0.5. **Threshold** is threshold value for the detection confidence of detected objects. Any objects with confidence value less than it will not be detected and displayed.
* "**BatchSize**": Type: Int. Default: 64. **BatchSize** is the number of training samples to be used in each iteration of inferencing the model. If an out of memory error is encountered, lower this value.
* "**ExtractBands**": Type: List of Ints or Strings. Default: Use all bands. **ExtractBands** is the array of band indexes or band names to extract from the input imagery. For example, [0,1,2].
* “**DataRange**”: Type: List of two numbers. **DataRange** is the range of data values if scaling or normalization was done in preprocessing. This represents the minimum and maximum value the deep learning model expects. ArcGIS will rescale the input raster tile to this data range if this is defined by the actual range of the input raster. For example, [0, 127].
* More can be found under “Esri model definition file” section in <https://pro.arcgis.com/en/pro-app/latest/help/analysis/image-analyst/deep-learning-in-arcgis-pro.htm>

**Some helpful links:**

<https://github.com/Esri/raster-functions/wiki/PythonRasterFunction#getconfiguration>

<https://github.com/Esri/raster-deep-learning/blob/master/docs/writing_deep_learning_python_raster_functions.md>

<https://github.com/Esri/raster-deep-learning/blob/master/docs/writing_model_definition.md>

<https://pro.arcgis.com/en/pro-app/latest/help/analysis/image-analyst/deep-learning-in-arcgis-pro.htm>