



Modernizing National Government with GeoAI & ArcGIS

National Government Webinar Series

Today's Presenters



Mark Cygan
Director of National Mapping Solutions



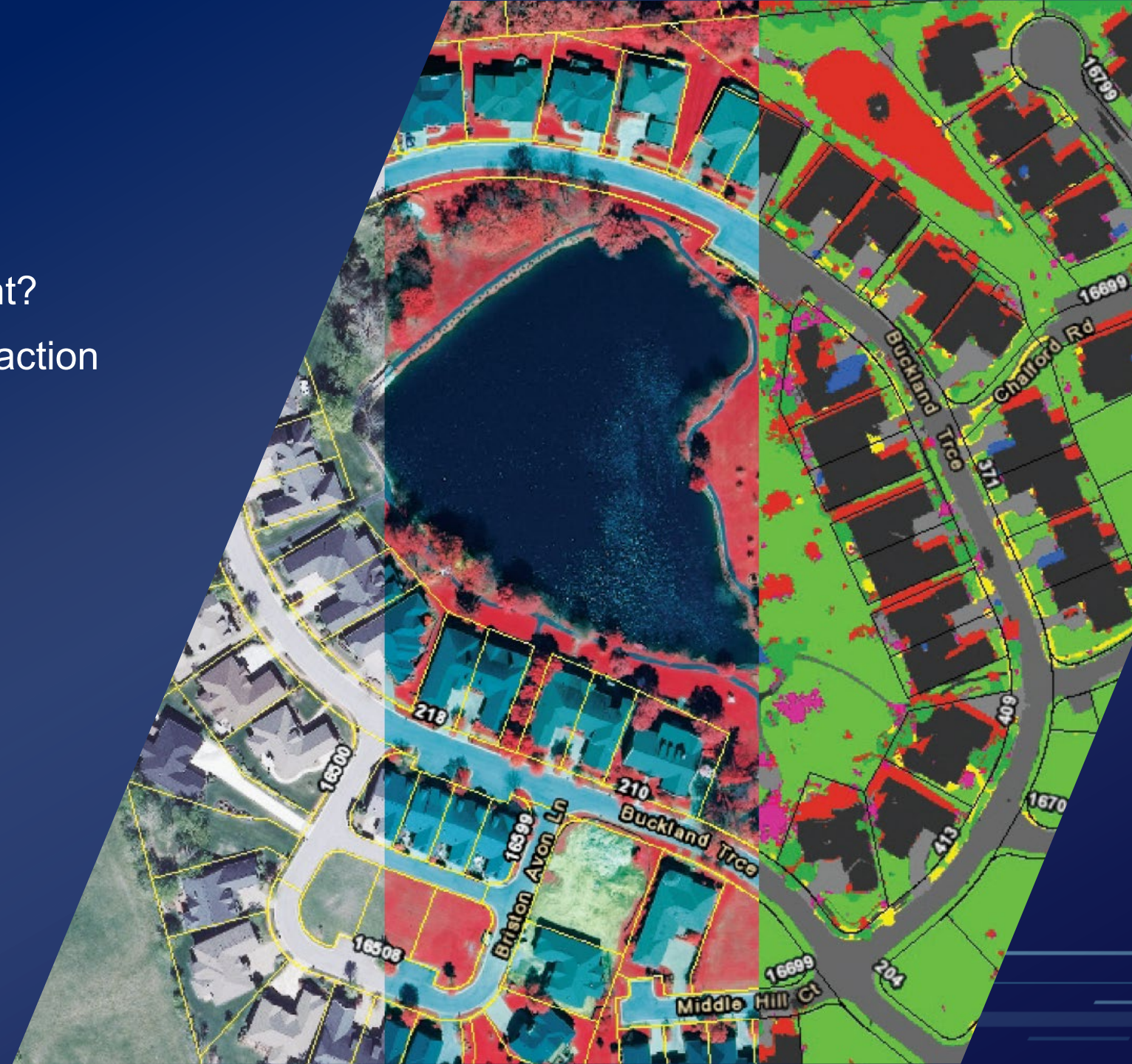
Linda Peters
Global Business Development Manager Statistics



Katherine Smyth
Solution Engineer Team Lead | National Government

Agenda

- Why GeoAI for National Government?
- Change Detection and Feature Extraction
- Demonstration
- GeoAI and Notebooks
- Demonstration
- Resources
- Open Q&A



GeoAI can help with

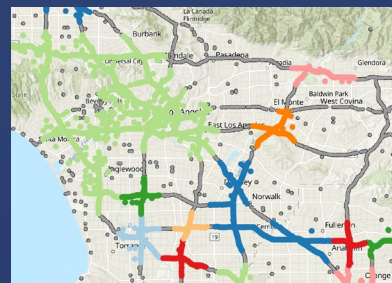
Prediction



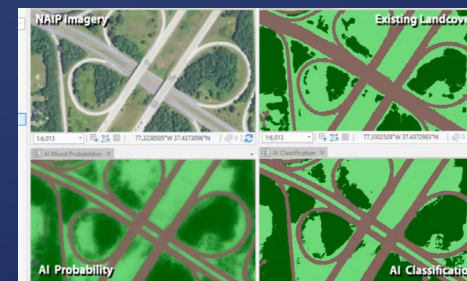
Object & Feature Detection



Clustering



Land Classification



Anomaly Detection



Challenges for National Government Organizations



- Expanding Stakeholder Expectations
- Escalating Demand for Broader Range of Information Products
- Delivering Near Real Time Information
- Increasing Demand for Better Resolution and Higher Quality Data
- Overall challenge of accuracy – v privacy



- Smaller Budgets and Revenues
- Less Staff and Resources
- Shrinking Timelines from Data Collection to Delivery
- Reduced Perception of Relevance
- Realities of Covid-19 and impacts on business continuity

GeoAI Benefits to National Government Organizations

Enabling Us to Do More with Less

- Responsiveness - Speeding Timely Information to Decision Makers
- Increasing Relevance - Expanding use of Technology, Data Analysis & Information Delivery
- Saving Time, Money, Labor and Environmental Impact
- Increasing Efficiency and Expanding Capacity
- Improving Accuracy and Engagement with User Community



GeoAI – Priorities with High Value to National Government

- **High-Value Workflows**

- **Change Detection of Areas & Features**
- **Feature Identification & Extraction**
- **Analysis and Assessment**

High Value Use Cases



Change Detection for Roads, Buildings and Land Cover



Road Extraction for National Mapping and Census Enumeration



Buildings for National Census Pre-Enumeration and National Mapping



National Land Cover & Agricultural Census

Imagery and Remote Sensing

ArcGIS is a comprehensive imagery platform

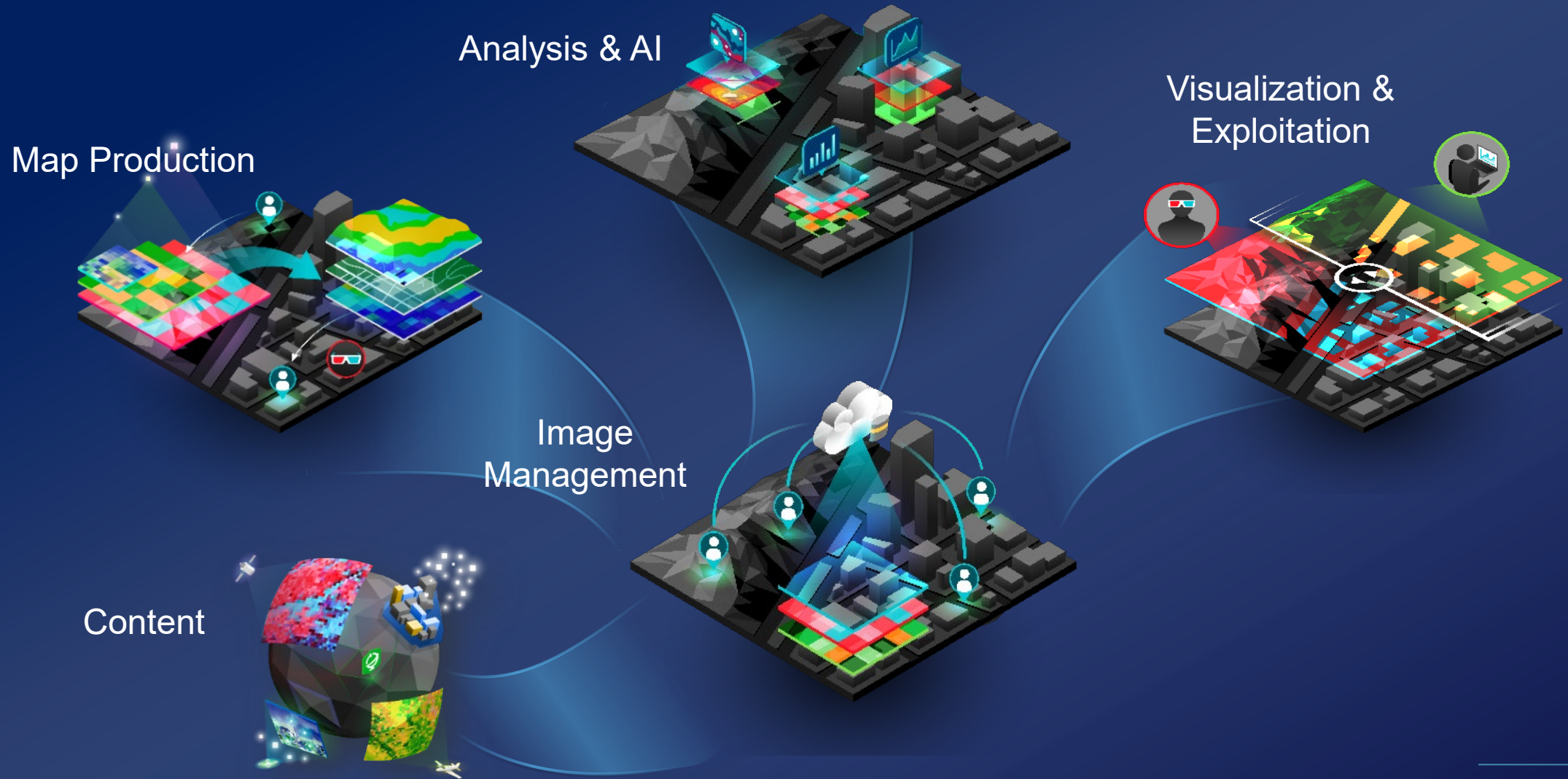


Image Analysis Extracting Information From Imagery

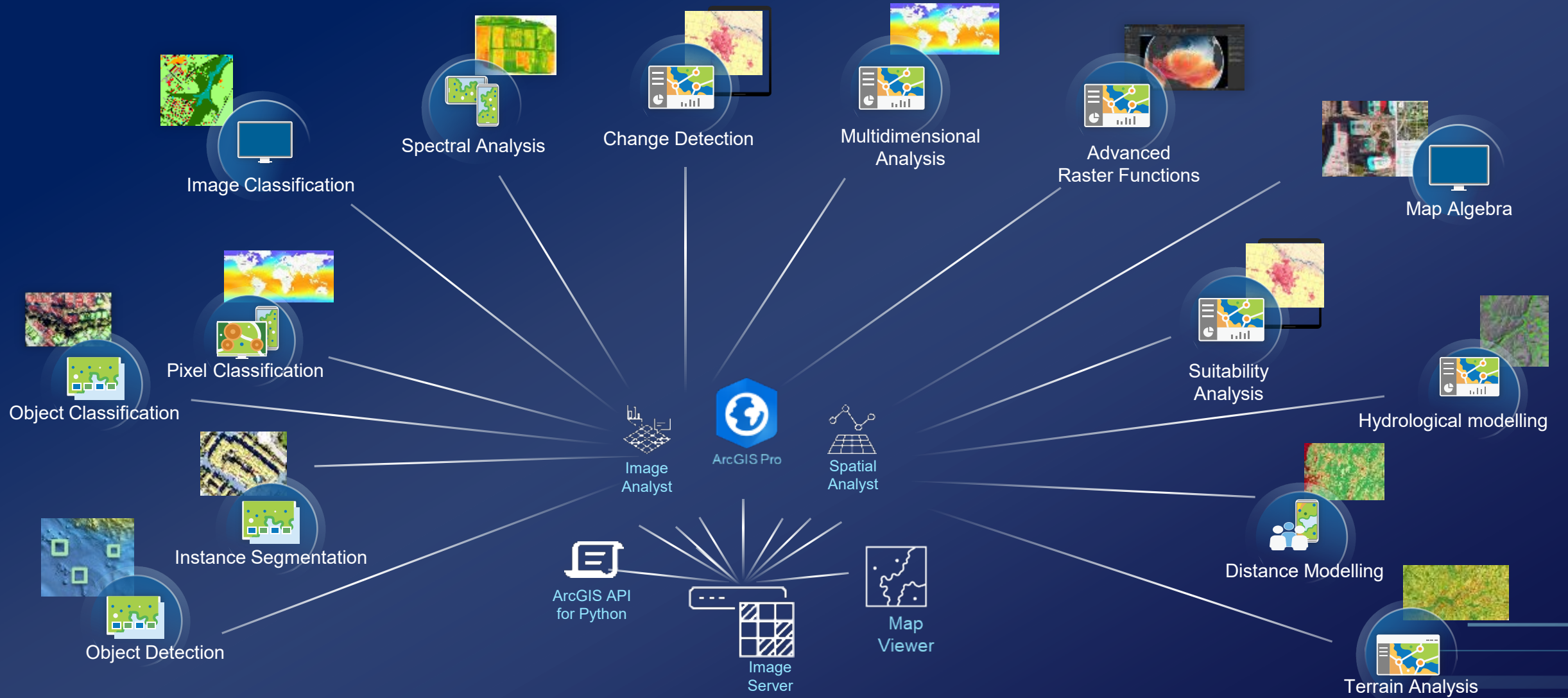


Image Analysis Extracting Information From Imagery



Artificial Intelligence

The diagram consists of three concentric circles. The outermost circle is the largest and contains the text 'Artificial Intelligence'. Inside it is a smaller circle containing 'Machine Learning'. Inside that is the smallest circle containing 'Deep Learning'. The circles are nested, indicating that Deep Learning is a subset of Machine Learning, which is a subset of Artificial Intelligence.

Machine
Learning

Deep
Learning

Deep Learning in ArcGIS



Integration



ArcGIS Deep Learning Workflow

Training Data Preparation

1

Create training samples

Label Objects for Deep Learning

2

Export images for training

Export Training Data for Deep Learning tool

Training

Object/Feature Classification

Instance Segmentation

3

Training a Model

Semantic Segmentation

Object Detection (including Change Detection)

Inferencing

Detect Objects

Classify Pixels

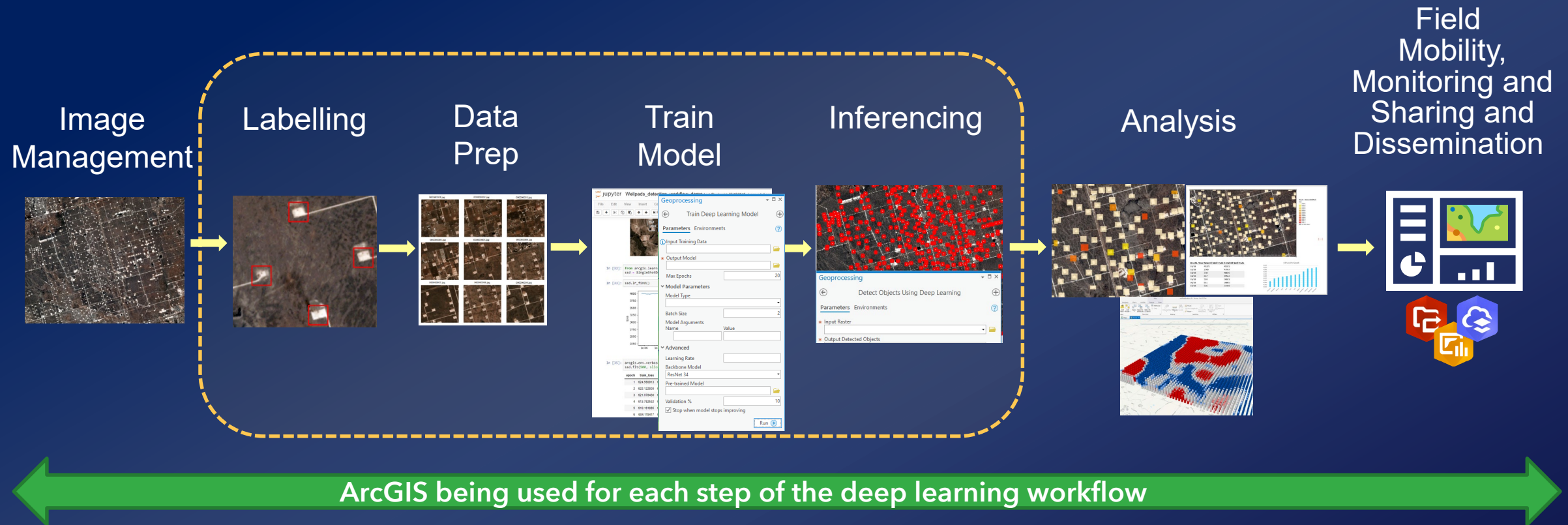
4

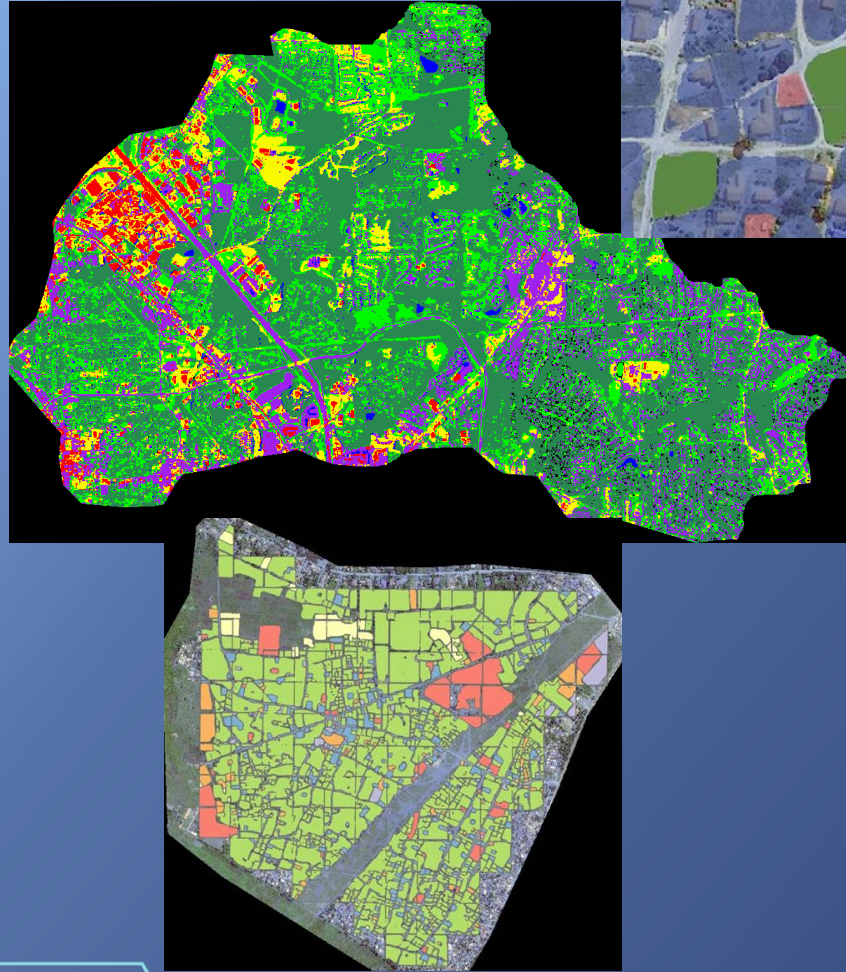
Inferencing

Classify Objects

Deep Learning Workflow in ArcGIS

End-to-end from raw imagery to structured information products





Demo

Land Cover Change Detection and
Map Feature Extraction
using Deep Learning

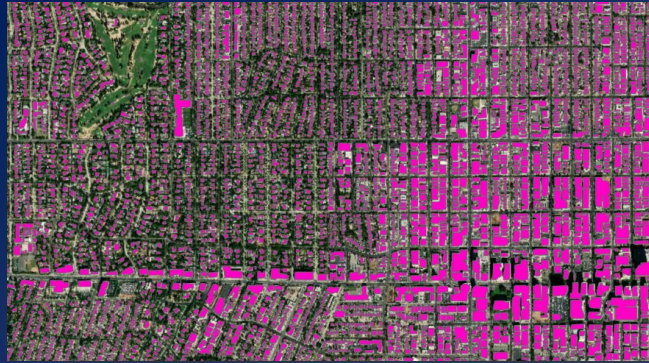
Examples for Imagery AI Workflows

Object Detection, Instance Segmentation, Land Cover, Change Detection..

Damaged Structures



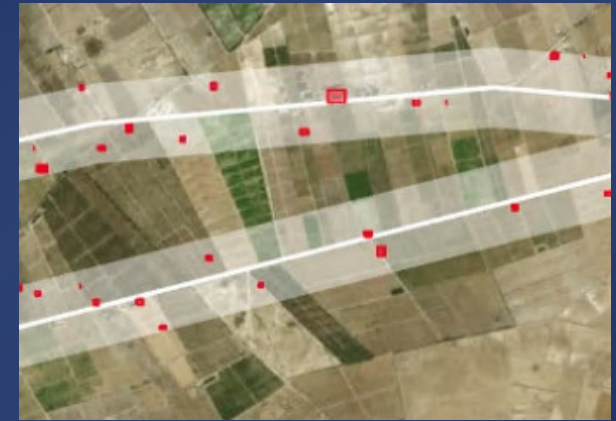
Building Footprints



Land Cover



Pipeline Encroachment



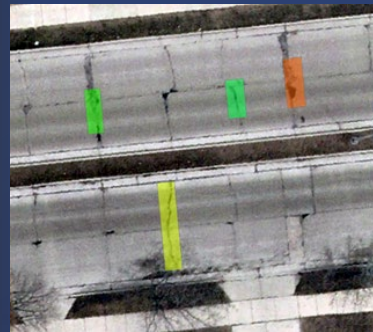
Roads



Oil Pads



Road Cracks



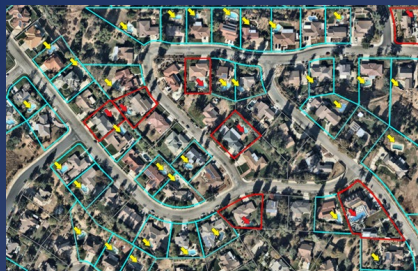
Cars



Palm Trees



Swimming Pools



Poll Question

Please Tick the Boxes that Apply to Your Organization

ArcGIS Notebooks sits at the intersection of ArcGIS and open data science



ArcGIS

- ArcGIS API for Python
- ArcPy



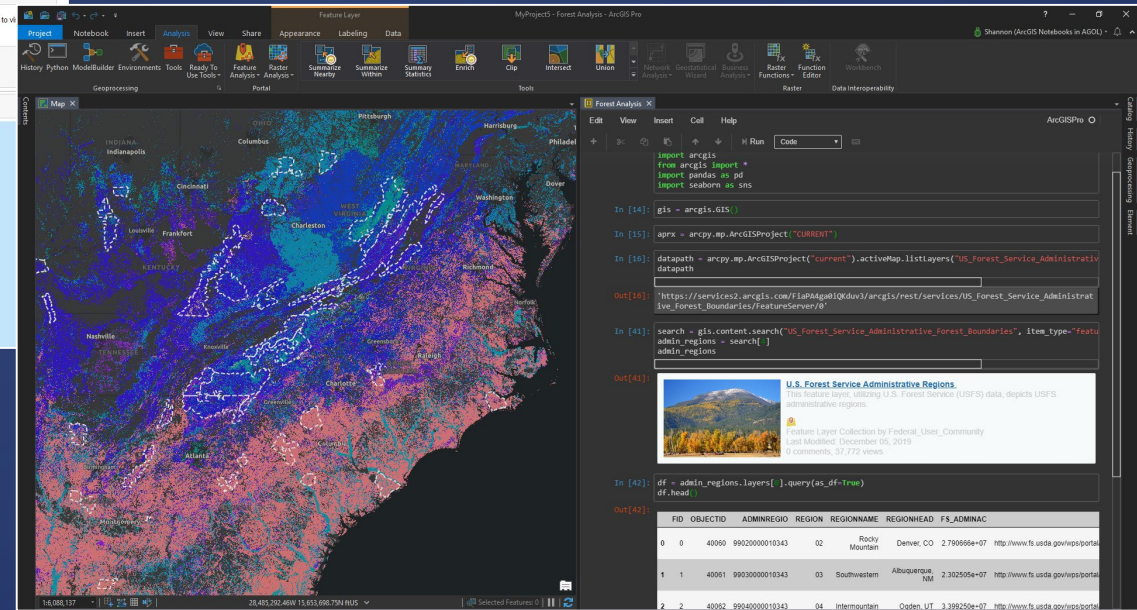
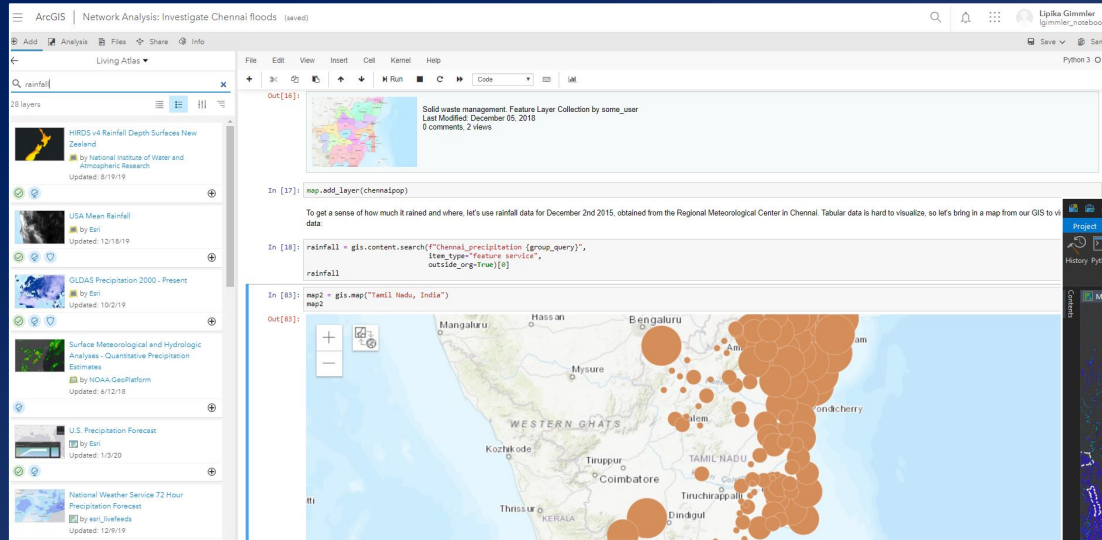
Open Source
Python & Data Science



ArcGIS Notebooks

A Spatially Optimized Jupyter experience within ArcGIS

Integrated Python notebooks in ArcGIS Pro allow you to seamlessly move data and analysis results between both.



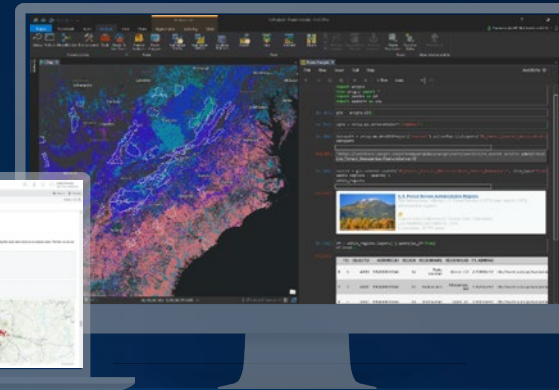
Hosted notebooks allow for easy searchability of datasets and to bring in analysis tools as code snippets.

Jupyter Notebook Integration for Modeling and Automation

ArcGIS Notebooks

ArcGIS Pro

ArcGIS Online



New & Improved

- Pro & Online
- Scheduling
- Parameterization
- Deep Learning Models
- Notebook Gallery

NEW

Pro

Integrated
Easy - No Setup

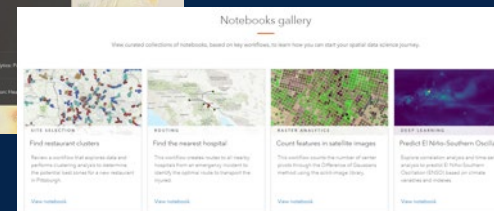
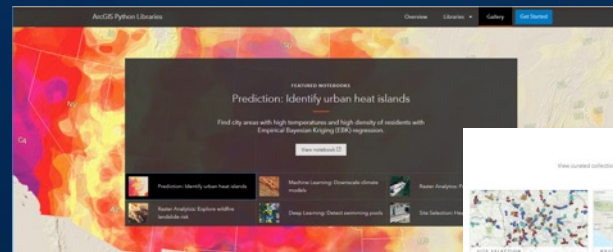
Enterprise

Specialized Server
Extensible

NEW

Online

Available as
an Extension



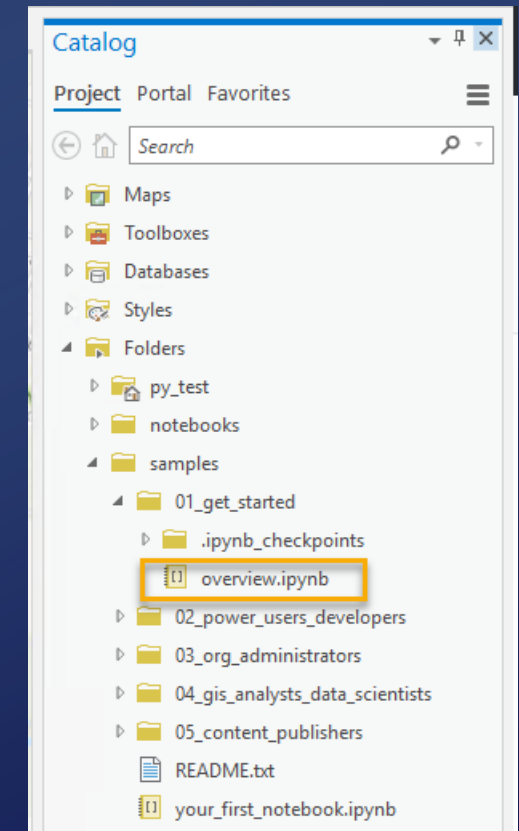
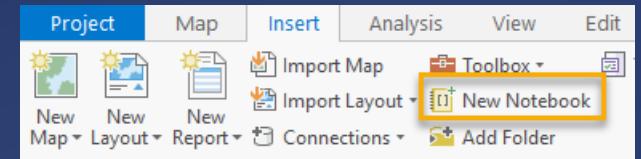
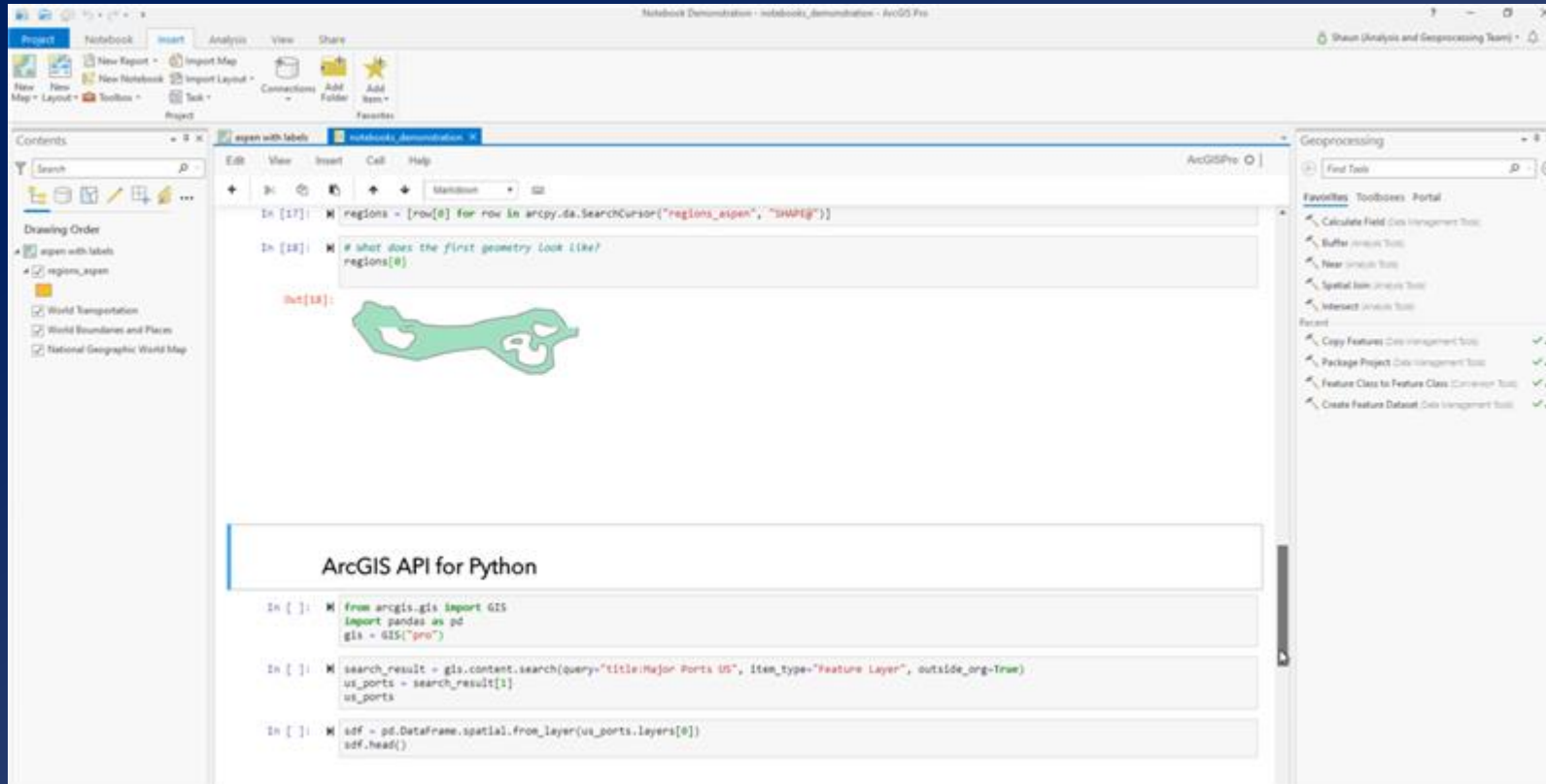
Notebook Gallery

Includes ArcGIS Python Libraries and
100s of Open Science Libraries

Scripting Processes and Workflows



ArcGIS Pro 2.5 seamlessly integrates Python Notebooks



ArcGIS Python Libraries

Python for Spatial Data Science

Convert and manage geographic data

Build machine and deep learning models

Perform advanced analytics

Automate workflows

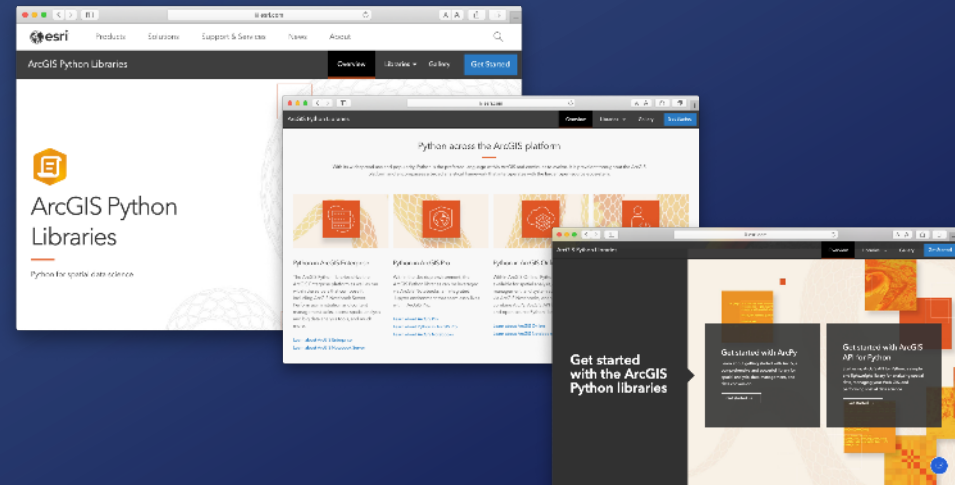


ArcGIS API for Python

Simple and lightweight library for analyzing spatial data, managing your Web GIS, and performing spatial data science.

ArcPy

Comprehensive and powerful library for spatial analysis, data management, and conversion.



Available across ArcGIS. . .
. . . In Pro, Enterprise, and Online

Deep Learning in ArcGIS API for Python (released version 1.8.1)

Make Deep Learning image analysis easier using `arcgis.learn` module



1. Export and Prepare Training Data

```
arcgis.learn.export_training_data  
arcgis.learn.prepare_data
```



2. Train Deep Learning Models

- **Object Detection:**
`arcgis.learn.SingleShotDetector`
`arcgis.learn.RetinaNet`
`arcgis.learn.MaskRCNN`
`arcgis.learn.FasterRCNN`
`arcgis.learn.YOLOv3`
- **Pixel Classification:**
`arcgis.learn.UnetClassifier`
`arcgis.learn.PSPNetClassifier`
`arcgis.learn.DeepLab`
- **Object Classification:**
`arcgis.learn.FeatureClassifier`
- **Others** (Text, Point Cloud, Tabular)



3. Run Inference at SCALE

```
arcgis.learn.detect_objects  
arcgis.learn.classify_objects  
arcgis.learn.classify_pixels
```



4. Model Management

```
arcgis.learn.list_models  
arcgis.learn.Model  
    Model.query_info  
    Model.install  
    Model.uninstall
```

More to come ...

ArcGIS API for Python

arcgis.learn module

The `arcgis.learn` module in ArcGIS API for Python enables Python developers and data scientists to easily train and use deep learning models with a simple, intuitive API.



Train Models

Before

```
labels = np.array(labels).ravel('labels')
print(labels.shape)

8. Convert Label to One Hot Vector
In [149]: print(labels.shape)
labels = labels.astype('category').reshape(-1, num_classes)
print(labels.shape)
labels = np.eye(num_classes)[labels]
print(labels.shape)

9. Define IoU Metric
In [150]: def mean_iou(y_true, y_pred):
    prec = []
    for i in np.arange(0.5, 1.0, 0.05):
        y_pred_ = tf.nn.round(y_pred * i)
        score_iou = tf.nn.mean(tf.nn.equal(y_true, y_pred_))
        K.get_session().run(tf.local_variables_initializer())
        with tf.control_dependencies([y_pred_]):
            score = tf.identity(score)
            prec.append(score)
    return K.mean(K.stack(prec), axis = 0)

10. Define Custom Loss Function
In [151]: class_weights = np.array([0.0000001, 1, 1, 1, 1, 1, 1])
weights = K.variable(class_weights)

def weighted_categorical_crossentropy(y_true, y_pred):
    # used predictions to get the class probs of each sample sum to 1
    y_pred = K.softmax(y_pred, axis = -1, keepdims = True)
    # clip to prevent NaN's and Inf's
    y_pred = K.clip(y_pred, K.epsilon(), 1 - K.epsilon())
    # calculate loss and weight loss
    loss = y_true * K.softmax(y_pred) * weights
    loss = -K.sum(loss, -1)
    return loss

11. Set Parameters before Training
In [149]: img_width = 256
img_height = 256
img_channels = 4
num_classes = 7

Land Cover Classes
```

- Dozens of lines of Code
- Installing External DL Frameworks
- HARD!

After

```
Train SingleShotDetector Model

from arcgis.learn import SingleShotDetector

ssd = SingleShotDetector(data, grids=[9], zooms=[1.0], ratios=[[1.0, 1.0]])

ssd.fit(10, lr=slice(1e-3, 1e-2))
```

- 3-5 lines
- Simple or no Installation (ArcGIS Pro & Notebooks)
- EASY

ArcGIS API for Python

Not just “training”!

Data Preparation

```
arcgis.learn.export_training_data  
arcgis.learn.prepare_data
```

Training DL Models

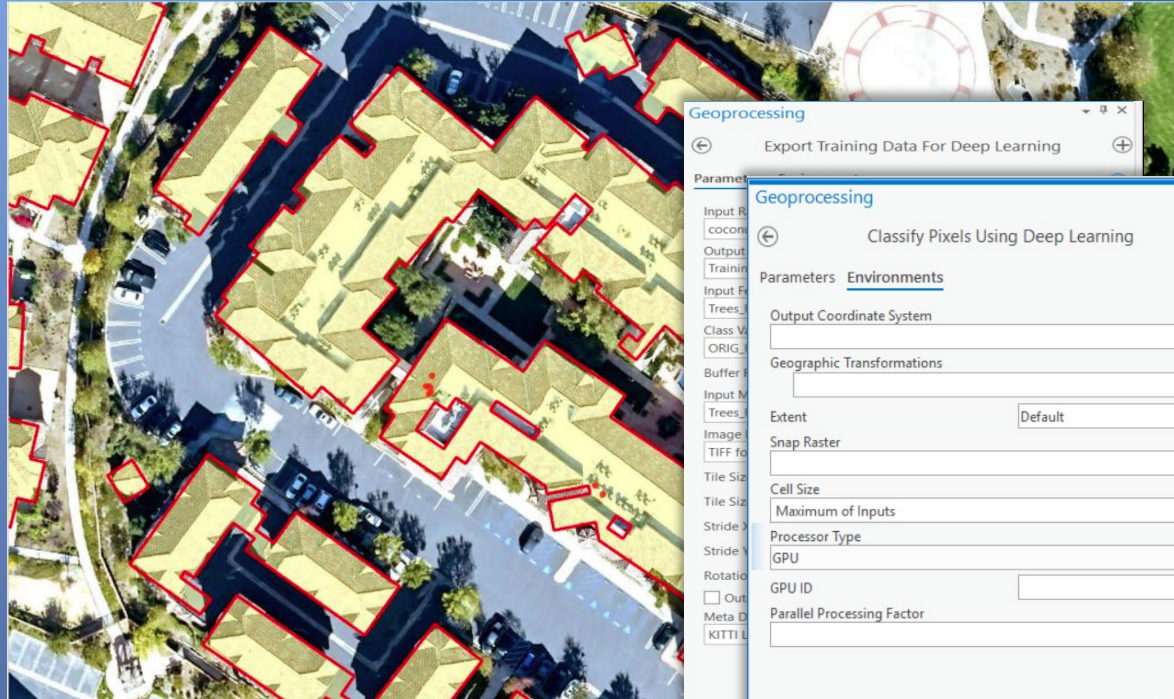
```
arcgis.learn.FeatureClassifier  
arcgis.learn.SingleShotDetector  
arcgis.learn.RetinaNet  
arcgis.learn.MaskRCNN  
arcgis.learn.UnetClassifier  
arcgis.learn.PSPNetClassifier  
arcgis.learn.DeepLab  
arcgis.learn.PointCNN  
arcgis.learn.EntityExtractor
```

Model Management

```
arcgis.learn.list_models  
arcgis.learn.Model  
    Model.install  
    Model.uninstall  
    Model.query_info
```

Inference APIs

```
arcgis.learn.detect_objects  
arcgis.learn.classify_pixels  
arcgis.learn.classify_objects
```



Geoprocessing

Export Training Data For Deep Learning

Geoprocessing

Classify Pixels Using Deep Learning

Parameters Environments

Output Coordinate System

Geographic Transformations

Extent Default

Snap Raster

Cell Size Maximum of Inputs

Processor Type GPU

GPU ID 0

Parallel Processing Factor

Run

Catalog Geoprocessing

Run

Parameter List:
Input R...
cocon...
Output...
Trainin...
Input F...
Trees...
Class V...
ORIG...
Buffer f...
Input M...
Trees...
Image...
TIFF fe...
Tile Siz...
Tile Siz...
Stride X...
Stride Y...
Rotatio...
Out...
Meta D...
KITTI L...

Demo

Notebooks and Deep Learning



Use-cases

Examples of GeoAI projects from the community





Manually created a high-resolution land cover map for precision conservation of the Chesapeake watershed

100k mi²

Area of watershed to map

2TB

File size of imagery to classify

18 months

Time to create map

By the time the land cover map was completed in December 2016, it was already out of date, and an update would be time-intensive and costly.

Land Classification Model

Algorithm Results

Working Platform: GeoAI Virtual Machine

Dataset: 100k mi² of imagery at 1-meter resolution, split in half geographically into train and test sets

Labeled Training Images

Chesapeake Conservancy Dataset

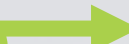
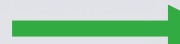


Convolutional Network Architecture
23 layer U-Net

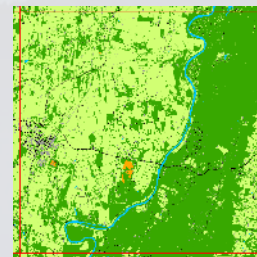
Test Images



Land Classification Model



Land Cover Map



91%

Average land classification accuracy

150

Hours taken, previously taken 2500

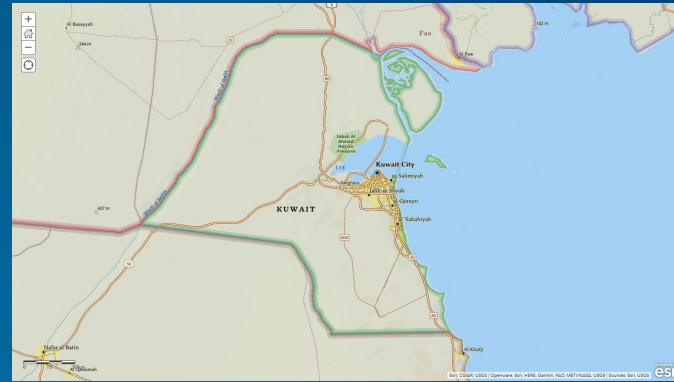
16x

Faster than Chesapeake Conservancy's previous methods

Kuwait



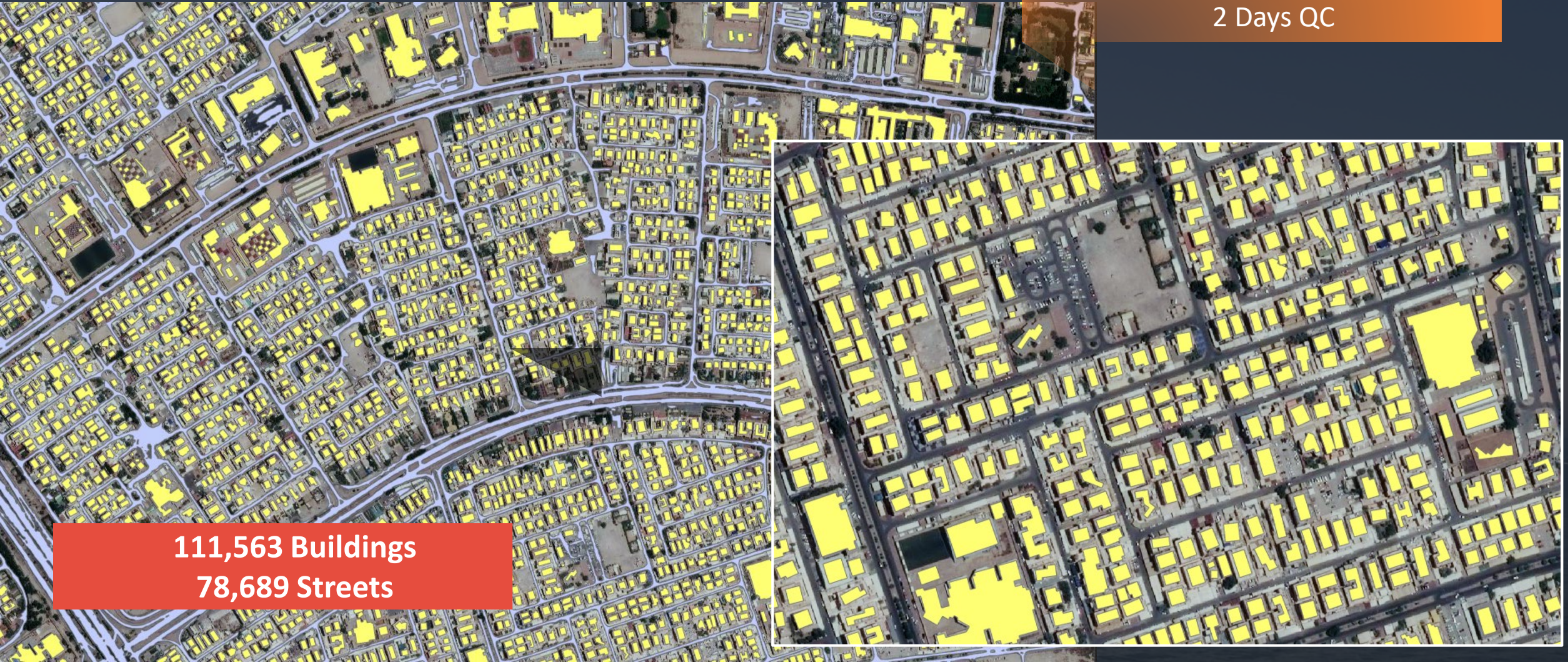
الهيئة العامة للمعلومات المدنية
The Public Authority For Civil Information



Output

1.5 hours processing
2 Days QC

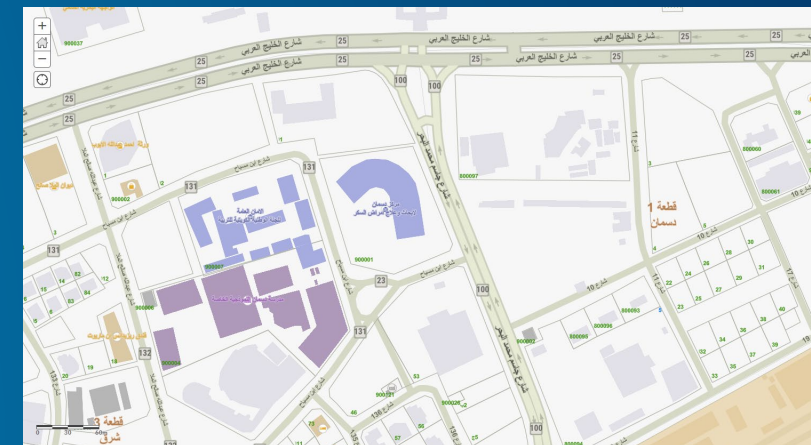
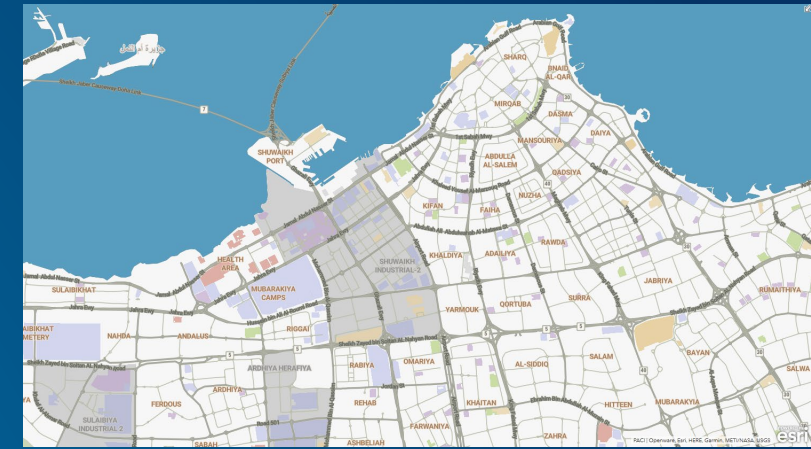
111,563 Buildings
78,689 Streets



PACI Kuwait and GeoAI



- More Responsive
- Improved Accuracy
- Savings in Time and Money
- Expanded Internal Capacity
- Maximized Return on Investment



Poll Question

Please Tick the Boxes that Apply to Your Organization

Where we offer machine learning integration.



ArcGIS API for Python

ArcGIS Velocity

ArcGIS Notebooks

ArcGIS Pro

ArcGIS Online

ArcGIS Enterprise

ArcGIS Hub - [Citizen Data Science](#)

ArcGIS QuickCapture - [Edge AI \(in R&D\)](#)

ArcGIS Insights

ArcGIS Pro for Intelligence

Resources

- **ArcGIS.Learn Documentation:** <https://developers.arcgis.com/python/api-reference/arcgis.learn.html>
- **Sample Notebooks** <https://developers.arcgis.com/python/sample-notebooks/>
- **ArcGIS API for Python** [Building Footprint Extraction using DL](#)
- **GeoAI blogs** <https://medium.com/geoai>
- **UC2020** [Geospatial Deep Learning with ArcGIS](#)
- **YouTube Videos:** Search for “ArcGIS Deep Learning”
- **7 Part Blog Series on GeoAI**
 - Future Impacts on Mapping and Modernization by GeoAI <http://alturl.com/chy5x>

Question & Answer

Please Enter Questions in the Questions Window



esri

THE
SCIENCE
OF
WHERE