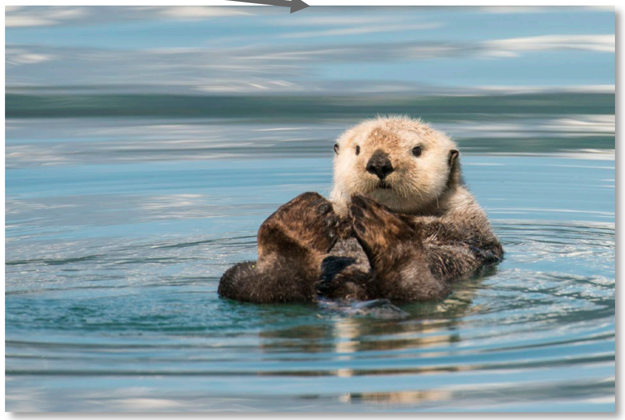
AI for Earth: Putting Azure to Work for Environmental Science

Dan Morris
Microsoft AI for Earth
dan@microsoft.com
aka.ms/aiforearth
aka.ms/dmorris

gratuitous pictures of animals that aren't related to my talk (coastal animals edition)



AI for Earth is three things…

Grants
aka.ms/ai4egrants

ML stuff
aka.ms/ai4etech

Data
aka.ms/ai4edata

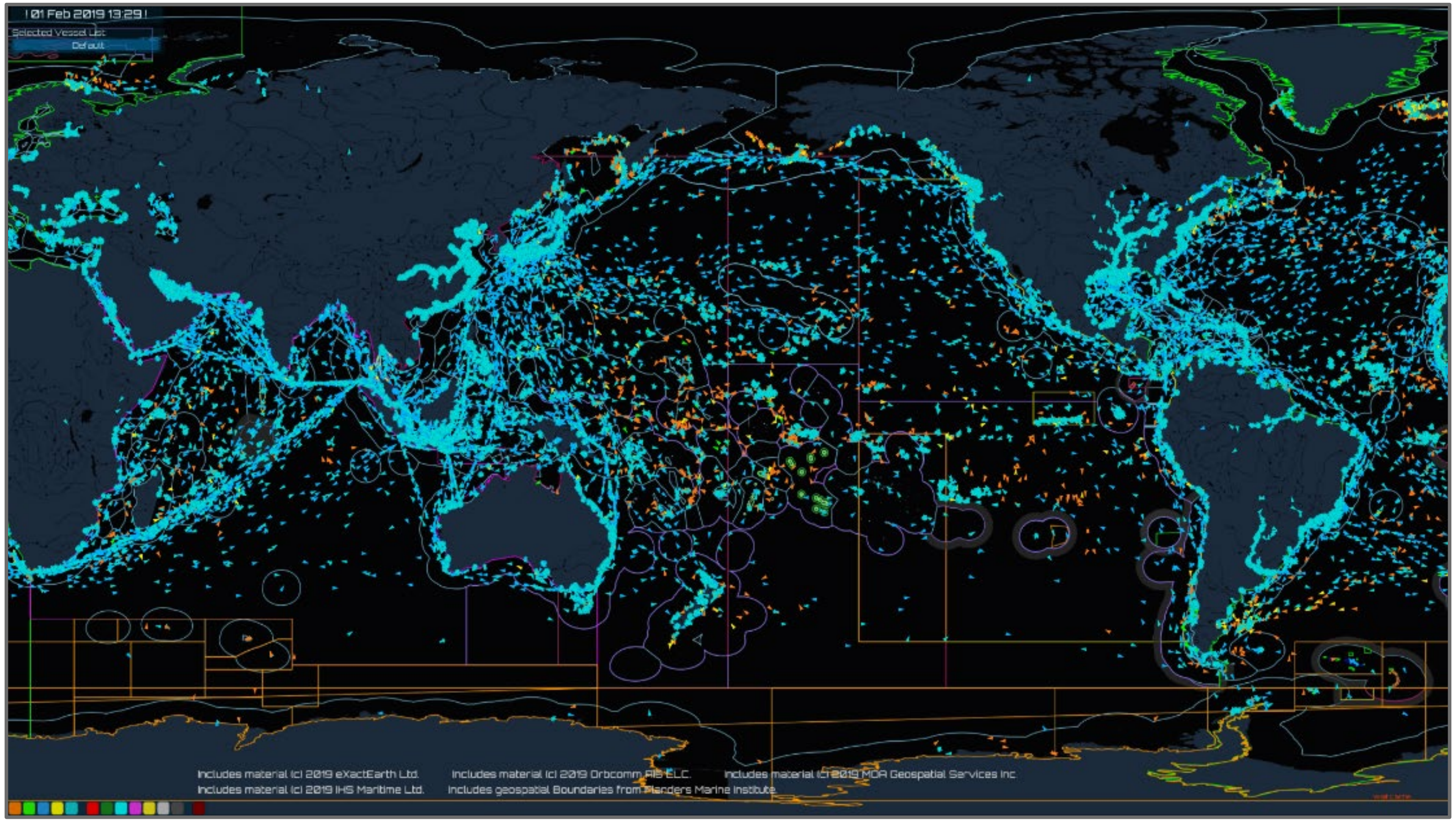
AI for Earth: our grants program

701
grants

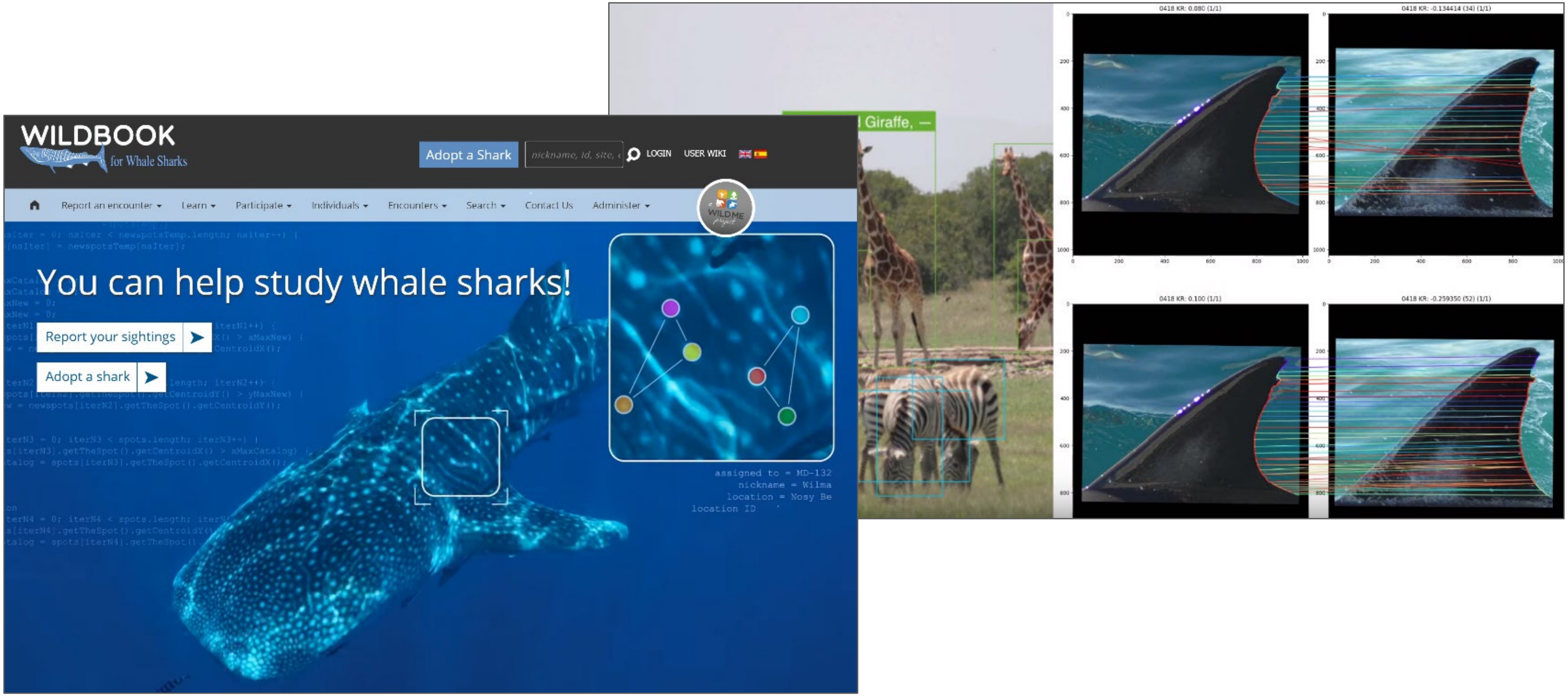
107
countries

aka.ms/ai4egrants

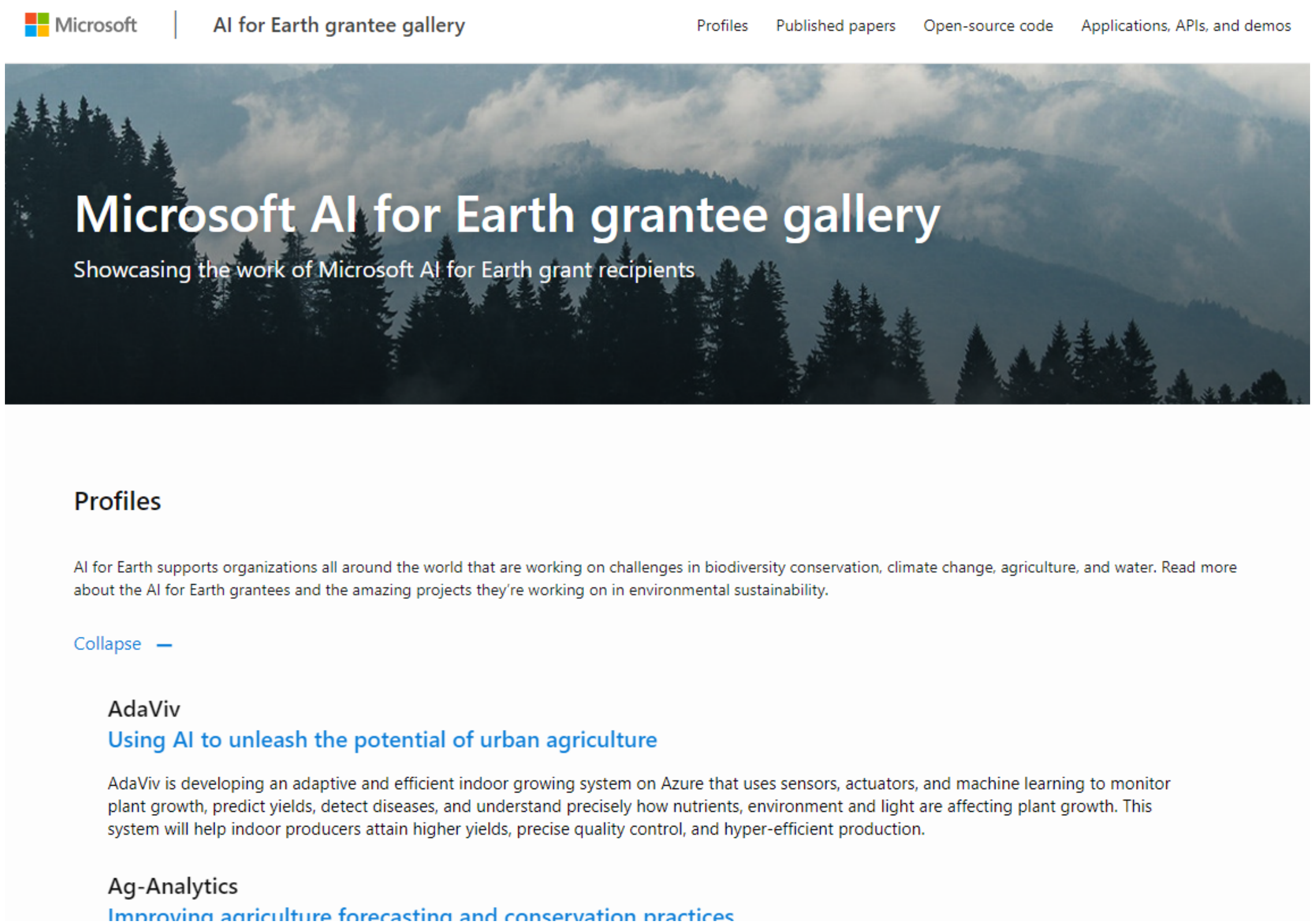
OceanMind : ML to detect illegal fishing



Wild Me: ML to scale scientific effort



AI for Earth: grantee stories


Microsoft | AI for Earth grantee gallery

Profiles Published papers Open-source code Applications, APIs, and demos

Microsoft AI for Earth grantee gallery

Showcasing the work of Microsoft AI for Earth grant recipients

Profiles

AI for Earth supports organizations all around the world that are working on challenges in biodiversity conservation, climate change, agriculture, and water. Read more about the AI for Earth grantees and the amazing projects they're working on in environmental sustainability.

Collapse —

AdaViv

Using AI to unleash the potential of urban agriculture

AdaViv is developing an adaptive and efficient indoor growing system on Azure that uses sensors, actuators, and machine learning to monitor plant growth, predict yields, detect diseases, and understand precisely how nutrients, environment and light are affecting plant growth. This system will help indoor producers attain higher yields, precise quality control, and hyper-efficient production.

Ag-Analytics

Improving agriculture forecasting and conservation practices

aka.ms/ai4egrantees

AI for Earth: partner stories

Ag-Analytics
Ag-Analytics combines data from farm equipment, satellite imagery, and weather forecasts in the cloud to provide precision recommendations to farmers.
Learn about Ag-Analytics >

Agrimetrics
The Agrimetrics Data Marketplace connects data and organizations across the food and farming sector to help create a more productive and sustainable food system.
Learn about Agrimetrics >

BasinScout Platform
The Freshwater Trust and Upstream Tech developed BasinScout® Platform to help identify the best places to improve groundwater and surface water quantity and quality.
Learn about BasinScout >

OceanMind
OceanMind uses satellites, vessel tracking data, and AI to reduce illegal and unreported fishing – preserving biodiversity and protecting livelihoods.
Learn about OceanMind >

OOICloud
The OOICloud project shares data from the Ocean Observatories Initiative with the scientific community through an open platform that helps ocean researchers study the impacts of climate change.
Learn about OOICloud >

Breeze Technologies
Breeze Technologies develops compact air quality sensors that use cloud and AI technology to collect data in real time, creating hyperlocal maps to better understand and improve air quality.
Learn about Breeze Technologies >

Cloud Agronomics
Cloud Agronomics uses remote-sensing technology and AI to provide growers with insights into their crops and soil, leading a new wave of proactive analytics to lower greenhouse gas emissions and spur sustainable food production.
Learn about Cloud Agronomics >

Conservation Metrics
Conservation Metrics combines machine learning, remote sensing, and scientific expertise to increase the scale and effectiveness of wildlife surveys.
Go to Conservation Metrics >

SunCulture
SunCulture develops irrigation and farming technology solutions to help smallholder farmers in Africa maximize yields and increase earnings.
Learn about SunCulture >

Terrafuse
Terrafuse leverages physics-enabled AI models to help organizations understand climate-related risk at the hyperlocal level.
Learn about Terrafuse >

eMammal
eMammal uses Microsoft Azure to store and organize images from citizen-run camera traps and complete projects aimed at learning more about the world's land mammals.
Learn about eMammal >

iNaturalist
iNaturalist is a platform that allows citizen scientists to contribute wildlife observations to environmental science research, using AI to accelerate the identification of thousands of species.
Go to iNaturalist >

NatureServe
NatureServe is leveraging Esri ArcGIS tools and Microsoft cloud computing to generate high-resolution habitat maps for imperiled species – providing critical information for smart conservation action.
Learn about NatureServe >

SilviaTerra
SilviaTerra is using aerial imagery and AI to survey forests at a national scale, transforming how conservationists and landowners measure and monitor forests.
Learn about SilviaTerra >

Wild Me
Wild Me develops open-source platforms for identifying and tracking wildlife, combining the strengths of AI and citizen scientists to fight extinction.
Learn about Wild Me >

aka.ms/ai4epartners

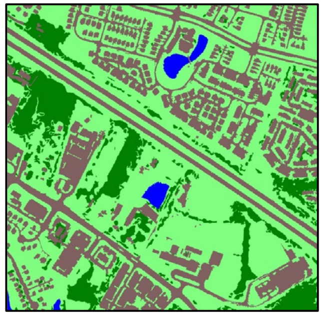
ML stuff AI for Earth builds:

Using ML to help conservation scientists spend *less time clicking stuff*, and *more time doing conservation*.

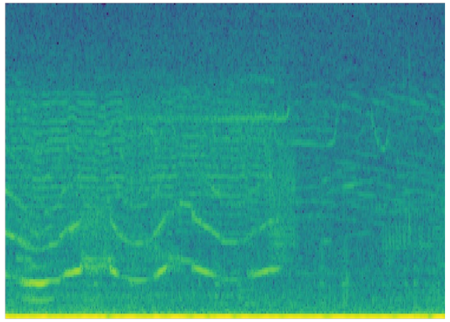
Accelerating *camera trap* surveys
github.com/microsoft/cameratraps

Accelerating *aerial* wildlife surveys
github.com/microsoft/aerial_wildlife_detection
github.com/microsoft/arcticseals

Accelerating *land cover* surveys
aka.ms/landcovermapping

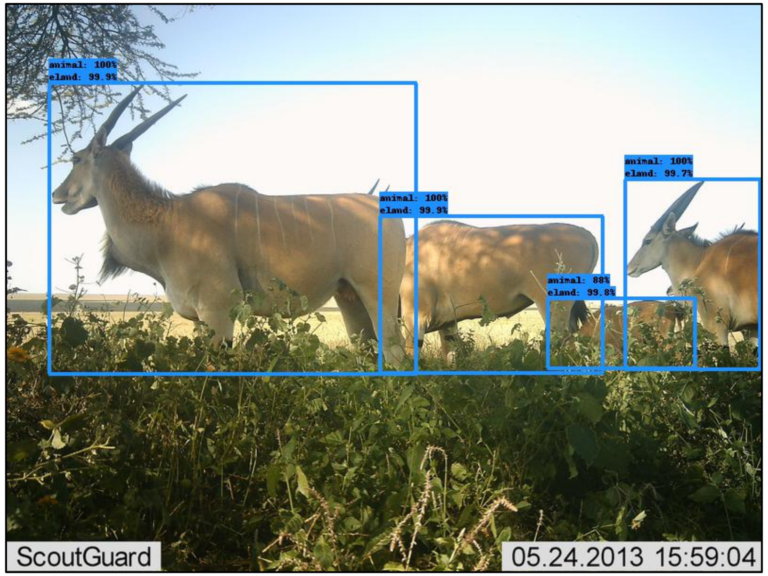


Accelerating *acoustic* wildlife surveys
github.com/microsoft/belugasounds
github.com/microsoft/multi_species_bioacoustic_classification



All the fun animal-related work in one place
aka.ms/biodiversitysurveys

Accelerating camera trap image processing



github.com/ microsoft /cameratraps

Demo

CLASSIFICATION RESULTS

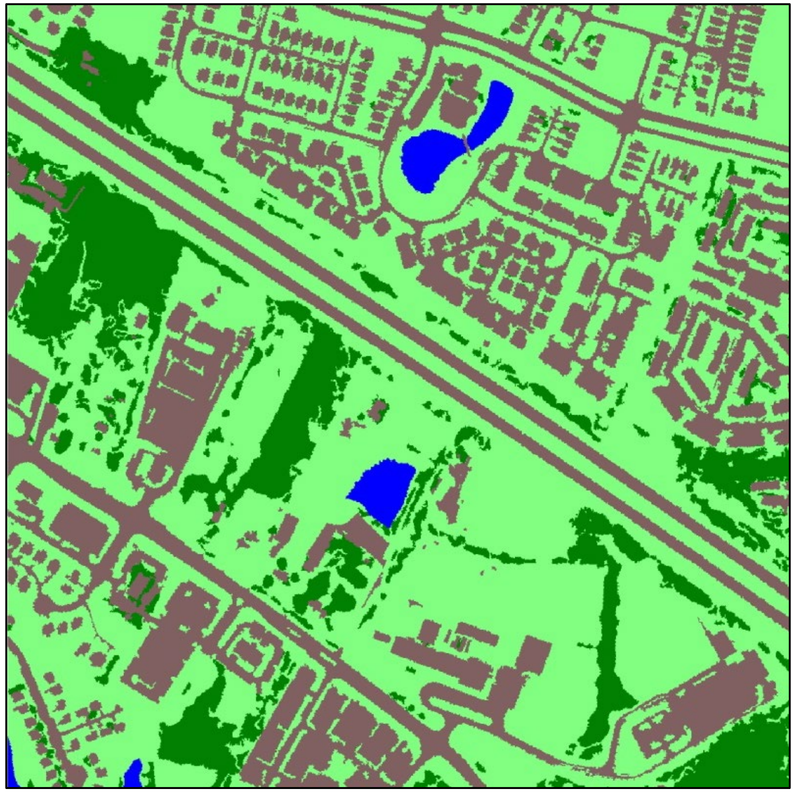
Animal 1

elephant

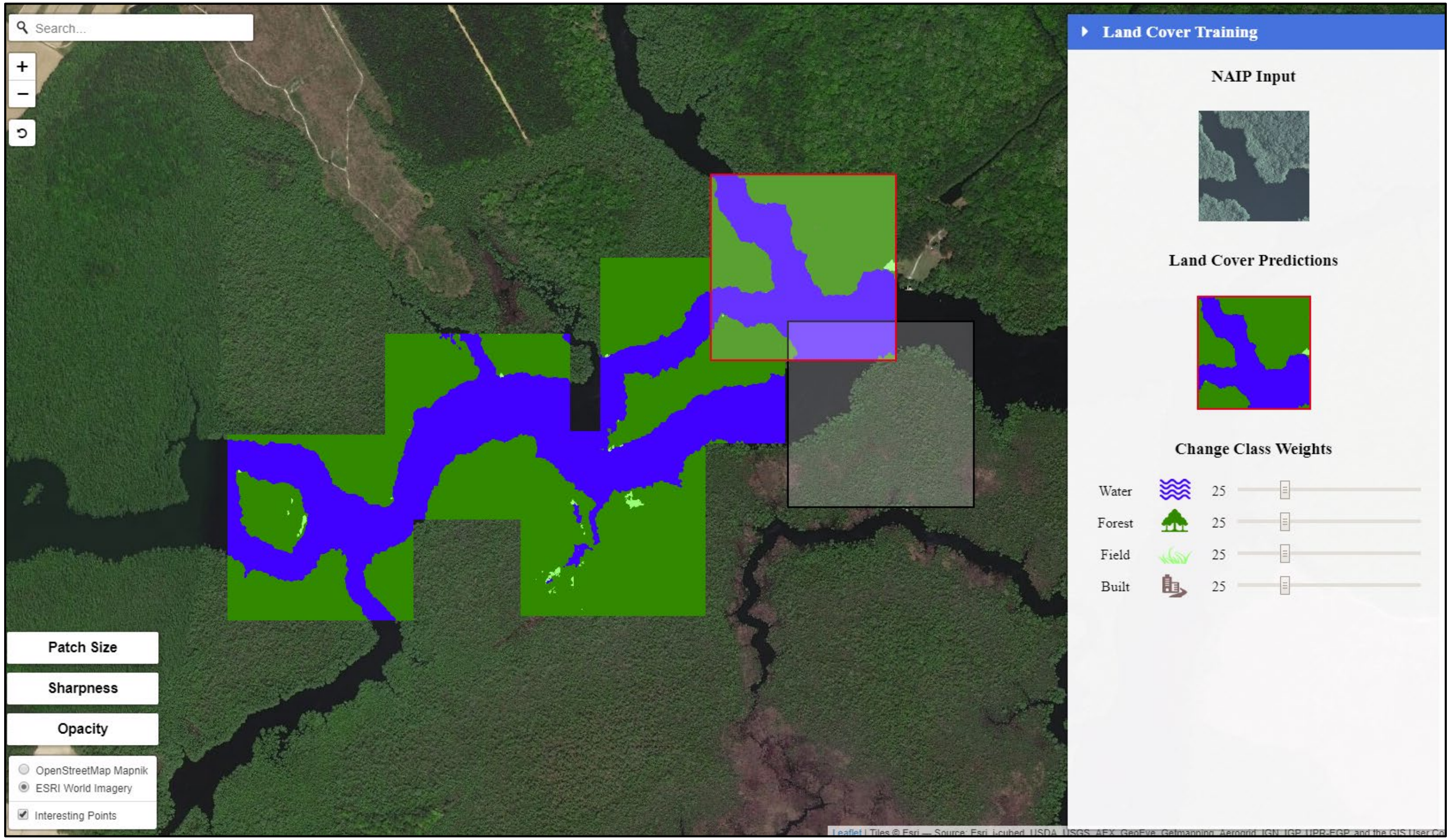
100.0%

aka.ms/ cameratrapdemo

Land Cover Mapping

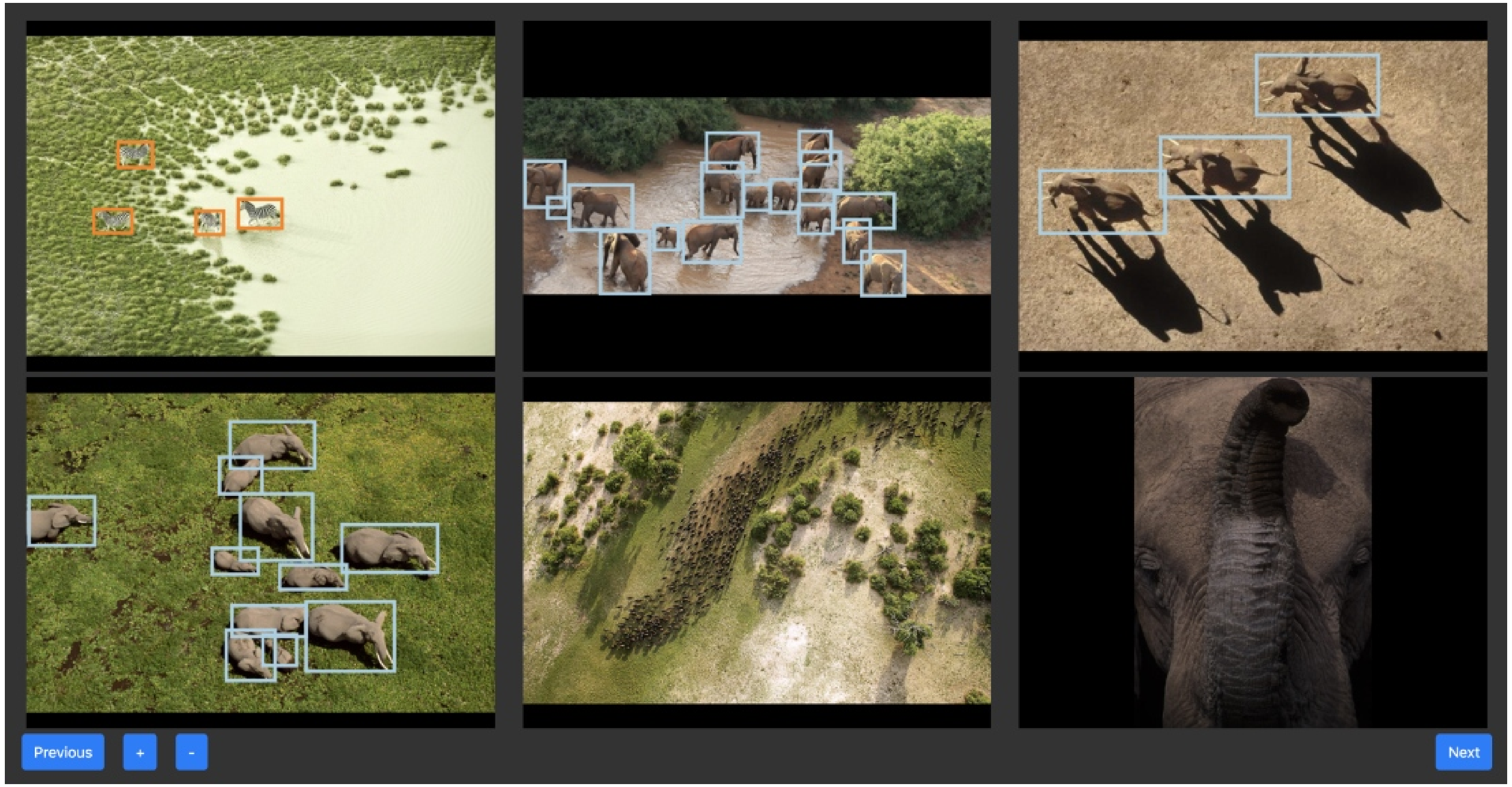


aka.ms/ landcovermapping

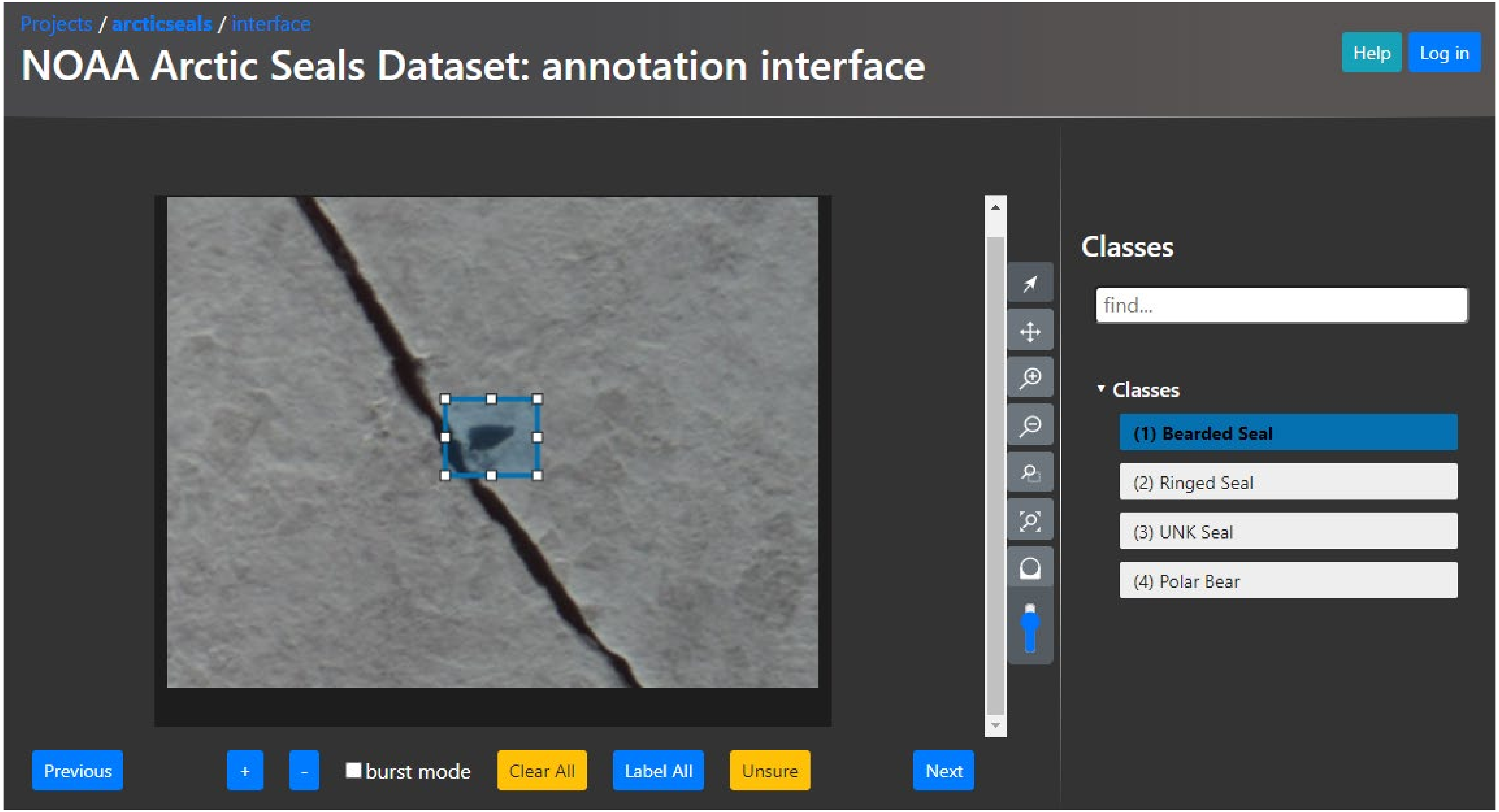


aka.ms/ landcoverdemo

Aerial wildlife detection
(but really active learning for geospatial object detection)

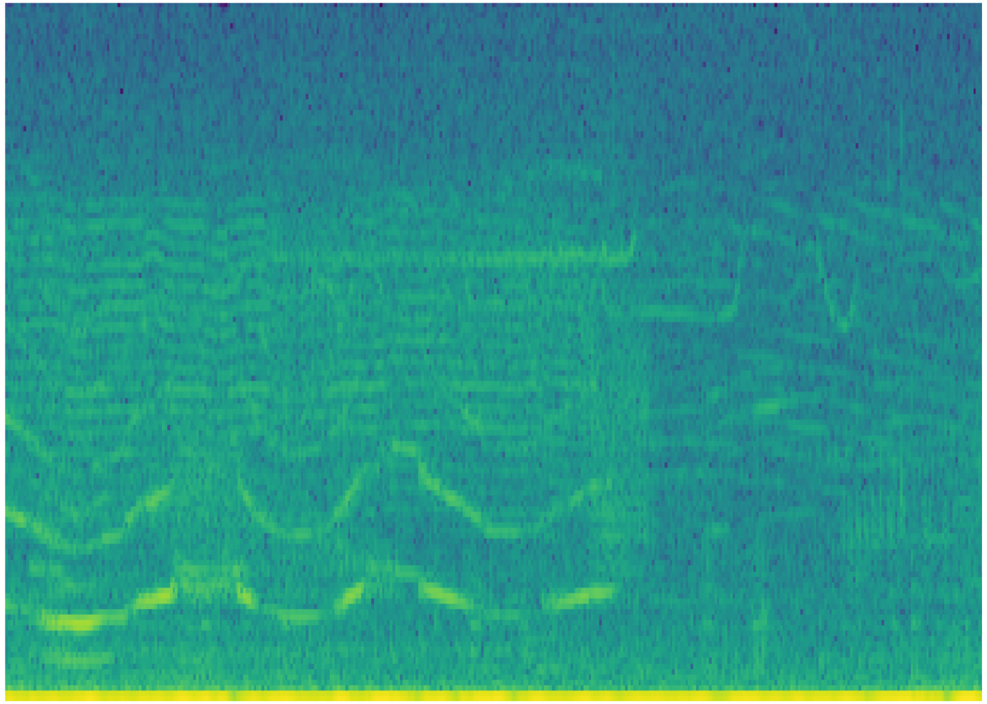


aka.ms/aerialwildlife

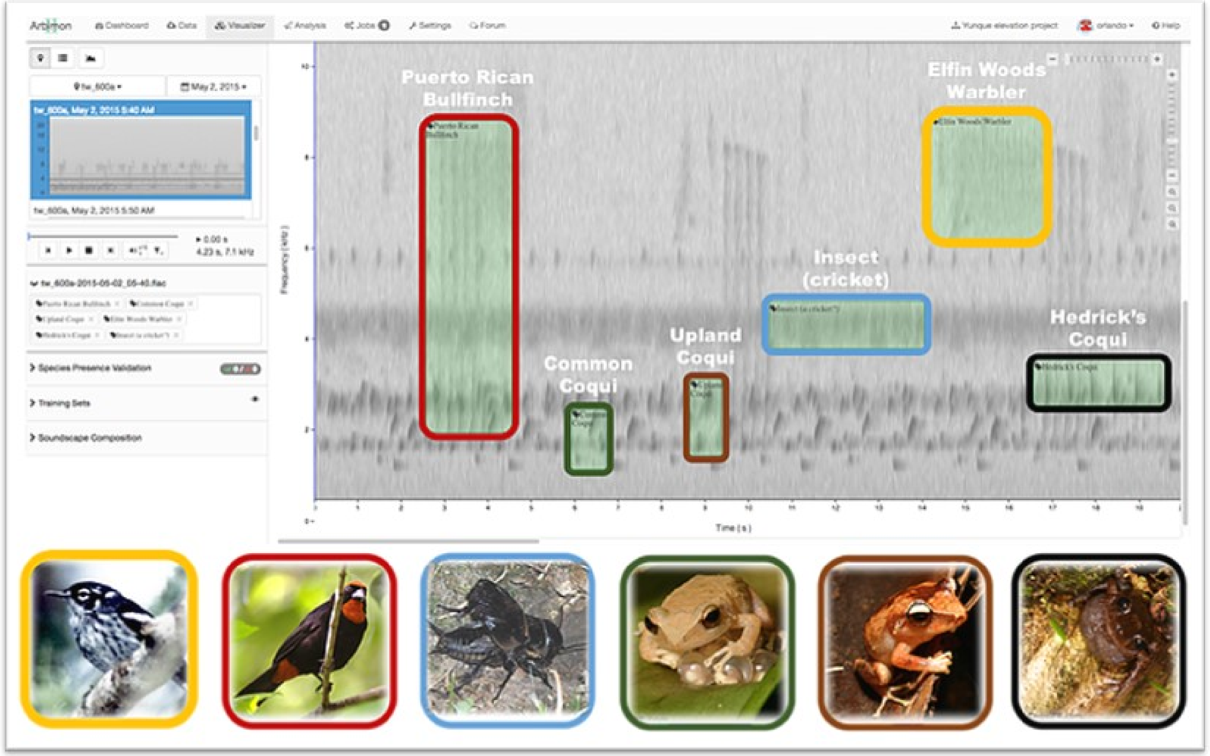


aka.ms/aerialwildlifedemo

Accelerating bioacoustic surveys

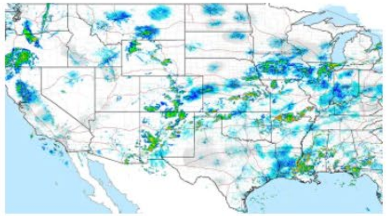


Accelerating beluga surveys
(w/NOAA Fisheries)
aka.ms/ belugasounds

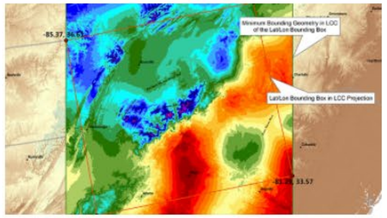


Accelerating rainforest soundscape surveys
(w/Sieve Analytics)
aka.ms/rainforest -acoustics

AI for Earth: data



NEXRAD
Radar data from 159 stations across the U.S.
NEXRAD on Azure Open Datasets >

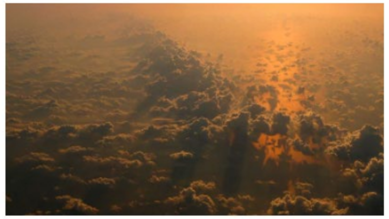


Daymet
Gridded temperature data across North America
Daymet on Azure Open Datasets >

NAIP
High-resolution aerial imagery across the continental U.S.
NAIP on Azure Open Datasets >

LILA BC
Labeled images related to wildlife conservation
Explore LILA BC >

NASADEM
Global topographic survey
NASADEM on Azure Open Datasets >



Global Hydro Estimator
Global precipitation estimates
GHE on Azure Open Datasets >

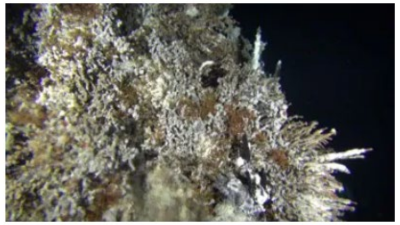
MODIS surface reflectance
500m-resolution global daily surface reflectance dating back to 2000
MODIS on Azure Open Datasets >

Harmonized Landsat Sentinel-2
30m-resolution satellite imagery for North America
HLS on Azure Open Datasets >

NOAA Global Forecast System
15-day US hourly weather forecast data
GFS on Azure Open Datasets >

NOAA Integrated Surface Data
Worldwide hourly weather history data
ISD on Azure Open Datasets >

GOES-16
Weather imagery of the Americas
GOES-16 on Azure Open Datasets >



Ocean Observatories Initiative CAMHD
High-resolution video from the floor of the Pacific Ocean
Video data from OOI >

aka.ms/ai4edata

AI for Earth: data

Demo notebook for accessing MODIS data on Azure

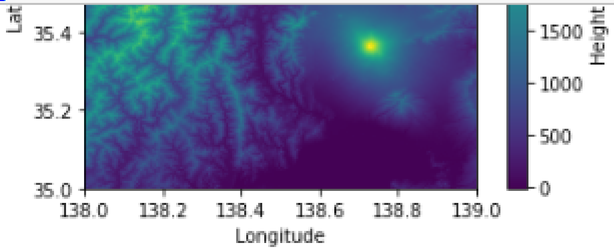
This notebook provides an example of accessing MODIS data from blob storage on Azure, including (1) finding the MODIS tile corresponding to a lat/lon coordinate, (2) retrieving that tile from blob storage, and (3) displaying that tile using the rasterio library.

This notebook uses the MODIS surface reflectance product as an example, but data structure and access will be the same for other MODIS products.

MODIS data are stored in the East US data center, so this notebook will run most efficiently on Azure compute located in East US. We recommend that substantial computation depending on MODIS data also be situated in East US. You don't want to download hundreds of terabytes to your laptop! If you are using MODIS data for environmental science applications, consider applying for an AI for Earth grant to support your compute requirements.

Imports and environment

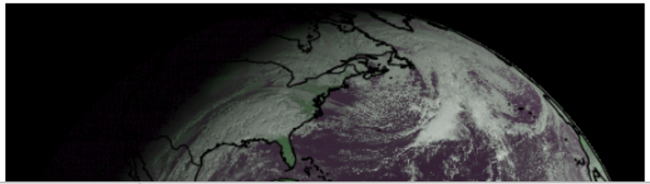
In [1]:
```
# Standard or standard-ish imports
import os
import tempfile
```



Also plot on a basemap

n [30]:
```
# To plot on a basemap, we'll want to downsample the data substantially
```

In [10]:
```
fig = plt.figure(figsize=(7.5, 7.5), dpi=100)

# This definitely looks slicker with fancy borders on, at the cost of some extra
# imports.
show_fancy_borders = True

if not show_fancy_borders:

    plt.imshow(rgb); ax = plt.gca(); ax.axis('off');

else:

    import metpy
    import cartopy.crs as ccrs

    # Pick an arbitrary channel to get the x/y coordinates and projection information
    # associated with the scan
    dummy_channel = dataset.metpy.parse_cf('CMI_C01')
    x = dummy_channel.x; y = dummy_channel.y

    ax = fig.add_subplot(1, 1, 1, projection=dummy_channel.metpy.cartopy_crs)
    ax.imshow(rgb, origin='upper', extent=(x.min(), x.max(), y.min(), y.max()))
    ax.coastlines(resolution='50m', color='black')
    ax.add_feature(ccrs.cartopy.feature.BORDERS);
```



Data on -boarded

- Sentinel -2 L2A
- Landsat 8
- NAIP
- MODIS surface reflectance
- NEXRAD L2
- Harmonized Landsat Sentinel -2
- NOAA GFS/ISD (and GFS warm -start ICs)
- NASADEM (SRTM)
- NOAA GHE
- GOES-16 (ABI-L1b-RadF and ABI -L2-MCMIPF)
- ASTER L1T (2000-2006)
- OOI CAMHD
- Met Office atmospheric ensembles

Data in progress

- Sentinel -1 GRD
- Sentinel -3 L2
- Sentinel -5P L2
- Landsat 4, 5, 7
- MODIS (~10 products)
- Two CMIP6 downscaled products
- NREL NSRDB, WAVE
- GDELT
- OpenStreetMap
- Expanded GOES -16/-17, GFS, ISD
- Lots and lots of smaller products:
    - Land cover: CCI, Corine, Copernicus, USGS Gap, NLCD
    - CDL, NED, 3DEP, WDPA, GPoW

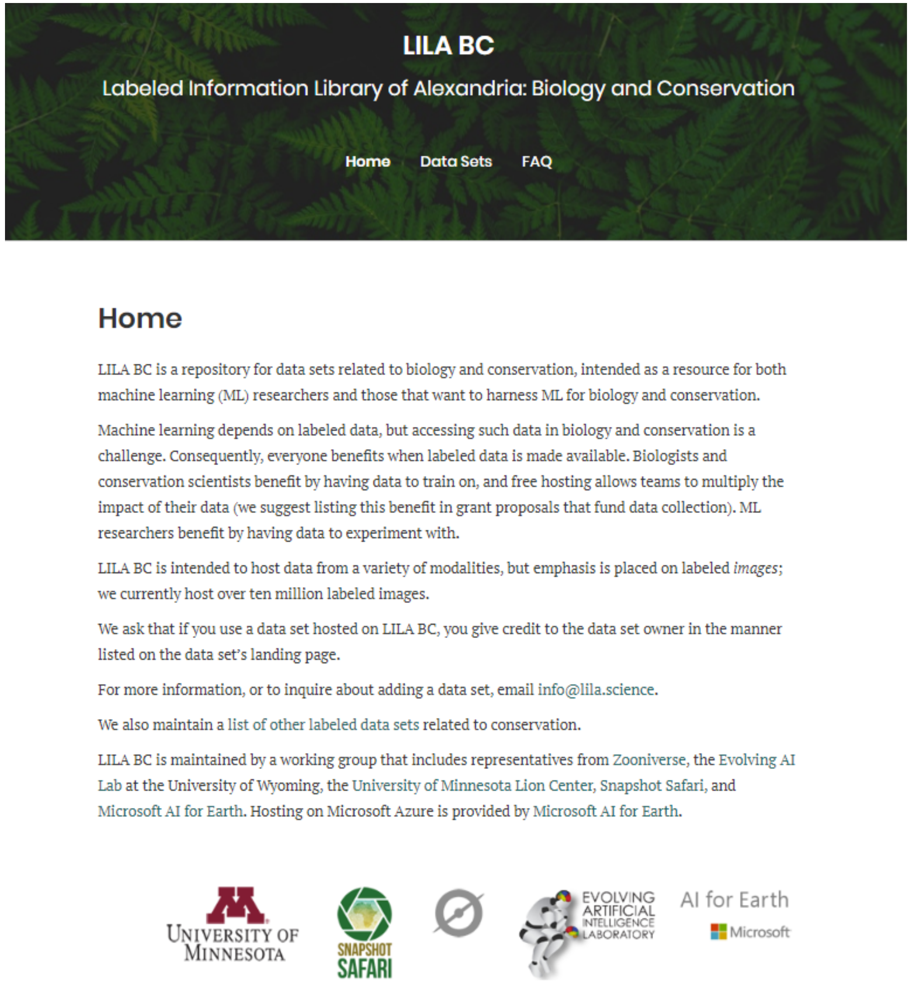
Big data: you can help us help you!

We want to know about…

- The public data sets you use
- The formats that delight/annoy you
- The tools you use to access/process data
- Your experiences with the data we host
- Your experiences with the data other clouds host

aiforearthdatasets@microsoft.com

And let’s make some noise for “small data”!


LILA BC

Labeled Information Library of Alexandria: Biology and Conservation

Home Data Sets FAQ

Home

LILA BC is a repository for data sets related to biology and conservation, intended as a resource for both machine learning (ML) researchers and those that want to harness ML for biology and conservation.

Machine learning depends on labeled data, but accessing such data in biology and conservation is a challenge. Consequently, everyone benefits when labeled data is made available. Biologists and conservation scientists benefit by having data to train on, and free hosting allows teams to multiply the impact of their data (we suggest listing this benefit in grant proposals that fund data collection). ML researchers benefit by having data to experiment with.

LILA BC is intended to host data from a variety of modalities, but emphasis is placed on labeled *images*; we currently host over ten million labeled images.

We ask that if you use a data set hosted on LILA BC, you give credit to the data set owner in the manner listed on the data set’s landing page.

For more information, or to inquire about adding a data set, email info@lila.science.

We also maintain a list of other labeled data sets related to conservation.

LILA BC is maintained by a working group that includes representatives from Zooniverse, the Evolving AI Lab at the University of Wyoming, the University of Minnesota Lion Center, Snapshot Safari, and Microsoft AI for Earth. Hosting on Microsoft Azure is provided by Microsoft AI for Earth.

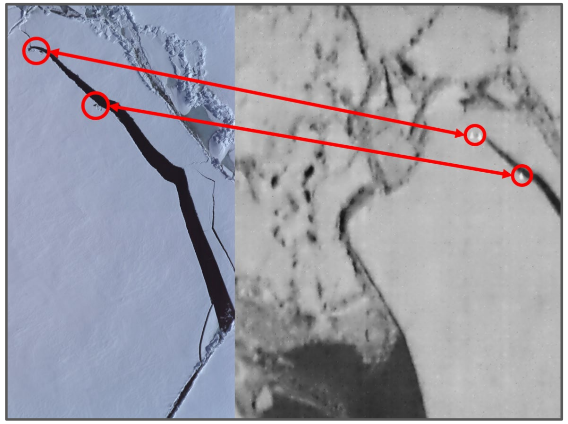
University of Minnesota

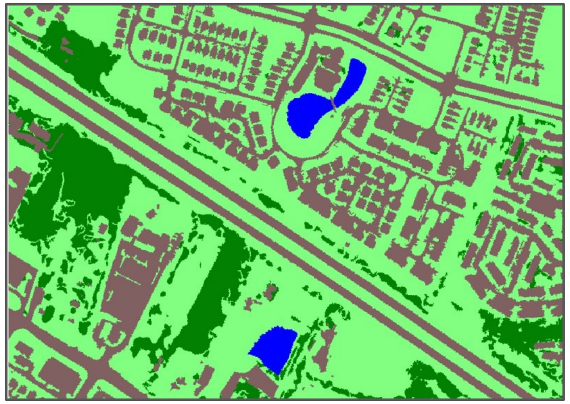
Snapshot Safari

Evolving Artificial Intelligence Laboratory

AI for Earth

Microsoft





http://lila.science

Apr 2020 ecosystems announcement

A healthy society requires a healthy planet

Apr 15, 2020 | Brad Smith - President

In January we launched Microsoft's carbon initiative, setting new goals for our company to become carbon negative by the end of this decade. While COVID-19 has upended daily life for almost all of us since then,

We'll protect more land than we use (how much land *do* we use?)

We'll build a "Planetary Computer" (a what?)

aka.ms/msft-ecosystems-announcement

The Planetary Computer

…a suite of data, tools, and applications that put planetary-scale geospatial data to work for environmental sustainability.

Why a Planetary Computer?

- Environmental sustainability depends on huge geospatial data sets, especially satellite imagery and climate data.
- Working with geospatial data is a pain unless you have a PhD in remote sensing.
- Working with very large data is a pain unless you have a PhD in distributed computing.
- Many of the people at the front lines of conservation have neither of the above, and they shouldn't have to.

The Planetary Computer *platform* puts key environmental datasets alongside processing tools and a managed compute environment to lower the access barrier for sustainability practitioners.

Planetary Computer *applications* put that platform to work for environmental decision-making.

Planetary Computer Components

- Geospatial **data** (~12PB by end of 2021)
- Data querying and processing **APIs**
- **Compute environment** for scientific workflows
- (Partner) **applications** that put all of the above to work for sustainability

Planetary Computer Data

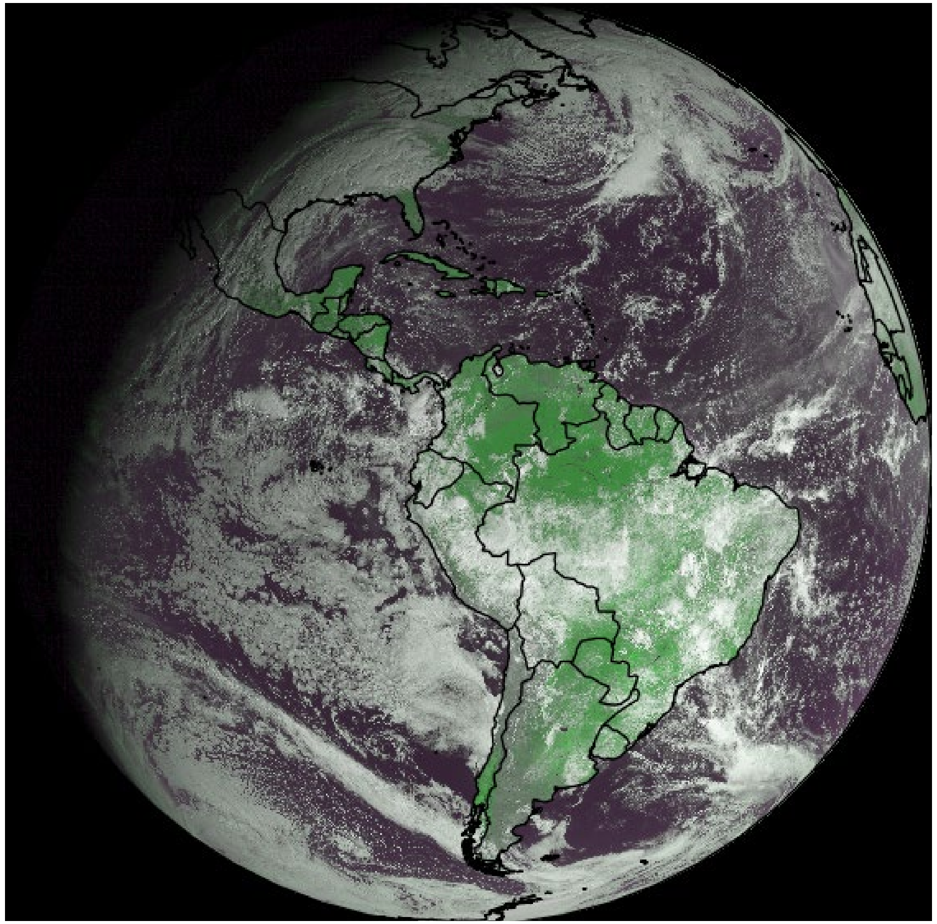
Remote sensing data
- Landsat 4, 5, 7, 8
- Sentinel-1, -2, -3, -5P
- GOES-16, -17
- MODIS, NAIP, ASTER

Weather/climate data
- CMIP6, ERA5, GFS, ISD, NEXRAD, GHE

Land cover data
- CCI, Corine, CCAP, NLCD, CDL, USGS GAP

External data sources
- Esri Living Atlas

Data discovery via a public catalog
- Published on GitHub Pages

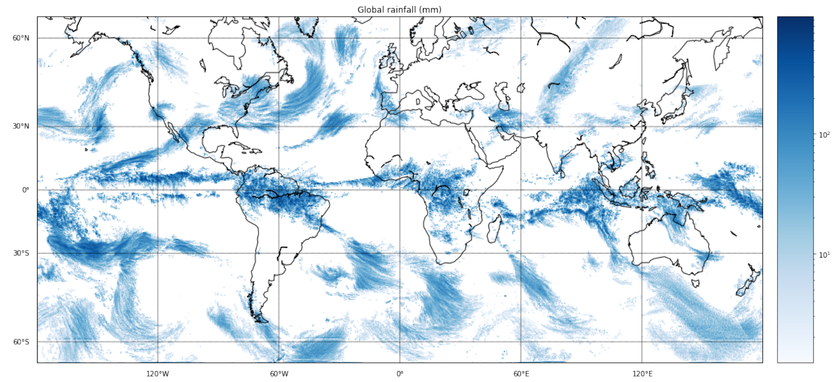


Planetary Computer APIs

Spatiotemporal queries for *images*
("find me all the *image files* that overlap with Wyoming in 2012")

Spatiotemporal queries for *pixels*
("I don't care about files, just give me all the *pixels* from 2012, cropped to Wyoming")

Image processing APIs: e.g. resampling to new grids, blending images



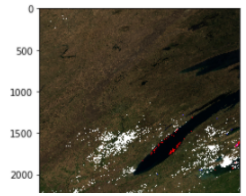
Planetary Computer Computing Environment

- Jupyter front-end
- Compute provisioning via JupyterHub
- Synchronous distributed processing
- Asynchronous distributed processing

```
    return destination_filename
```

Access and plot a MODIS tile

```
In [3]: # Files are stored according to:
#
# http://modissa.blob.core.windows.net/[product]/[htile]/[vtile]/[year][day]/filename

# Surface reflectance
product = 'MCD43A4'

# Let's look at the tile containing Chicago, IL, on May 15, 2019 (day of year 135)
h,v = lat_lon_to_modis_tile(41.881832,-87.623177)
daynum = '2019135'
folder = product + '/' + '{:0>2d}/{:0>2d}'.format(h,v) + '/' + daynum

# Find all .tiff files from this tile on this day, one file per channel
files = list_tiff_blobs_in_folder(modis_container_name,folder)

norm_value = 4000

# Channel 7 in a MCD43A4 file corresponds to MODIS band 1.
#
# Let's map bands 1, 4, and 3 (channels 7,10,9) to RGB.
channels = [7,10,9]
image_data = []
for ifn in channels:
    remote_fn = files[ifn]
    url = modis_blob_root + '/' + remote_fn
    fn = download_url(url)
    raster = rasterio.open(fn,'r')
    band_array = raster.read(1)
    raster.close()
    band_array = band_array / norm_value
    image_data.append(band_array)
rgb = np.dstack((image_data[0],image_data[1],image_data[2]))
np.clip(rgb,0,1,rgb)
plt.imshow(rgb)

Downloading file MCD43A4.A2019135.h11v04.006.2019149220457.hdf_08.tiff...done, 11546274 bytes.
Downloading file MCD43A4.A2019135.h11v04.006.2019149220457.hdf_11.tiff...done, 11546274 bytes.
Downloading file MCD43A4.A2019135.h11v04.006.2019149220457.hdf_10.tiff...done, 11546274 bytes.

Out[3]: <matplotlib.image.AxesImage at 0x223970bbc48>
```



AI for Earth:
Putting Azure to Work for Environmental Science

Dan Morris
Microsoft AI for Earth
dan@microsoft.com
aka.ms/aiforearth
aka.ms/dmorris