Accelerating Deep Learning Research with the NSF NRP Nautilus HyperCluster

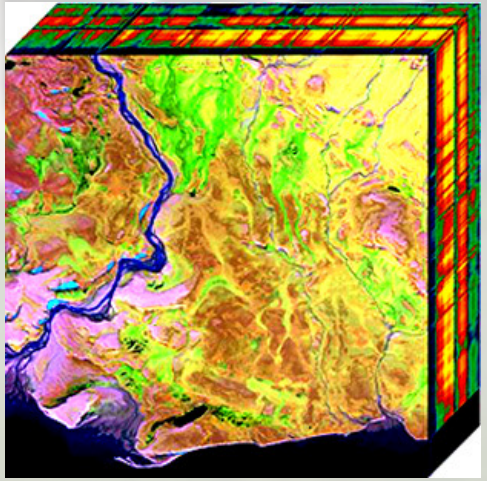
MORENet Technical Summit

20 Feb 2023

University of Missouri

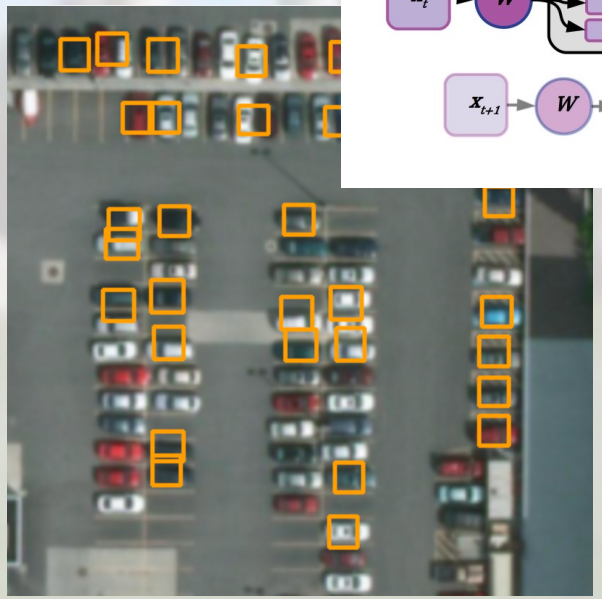
Problem Space: Deep Learning in Remote Sensing

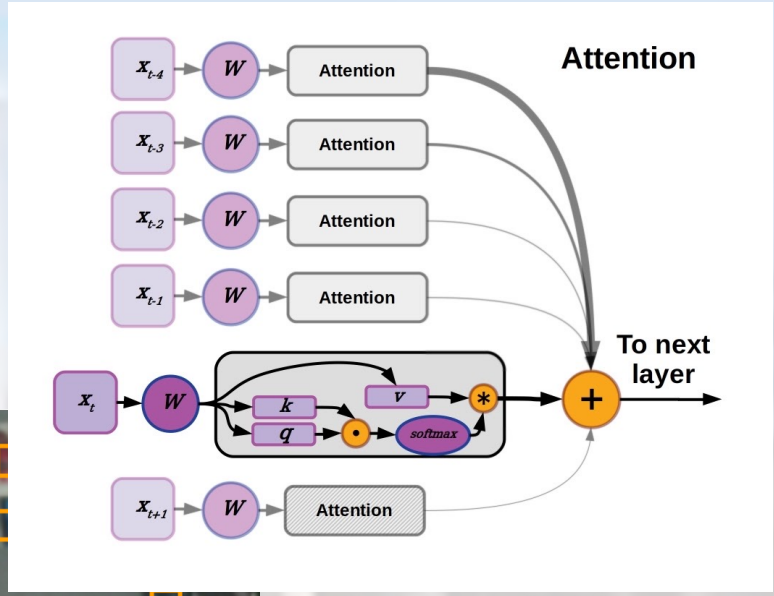
- Remote Sensing imagery sources:
  - Satellites
  - Unmanned Aerial Systems (UAS)
- Remote sensing is a Big Data problem!
  - Satellites collect imagery 24/7/365
    - Trillions of pixels per day
  - High levels of dimensionality available
    - Multispectral and hyperspectral imagery



Problem Space: Deep Learning & Remote Sensing

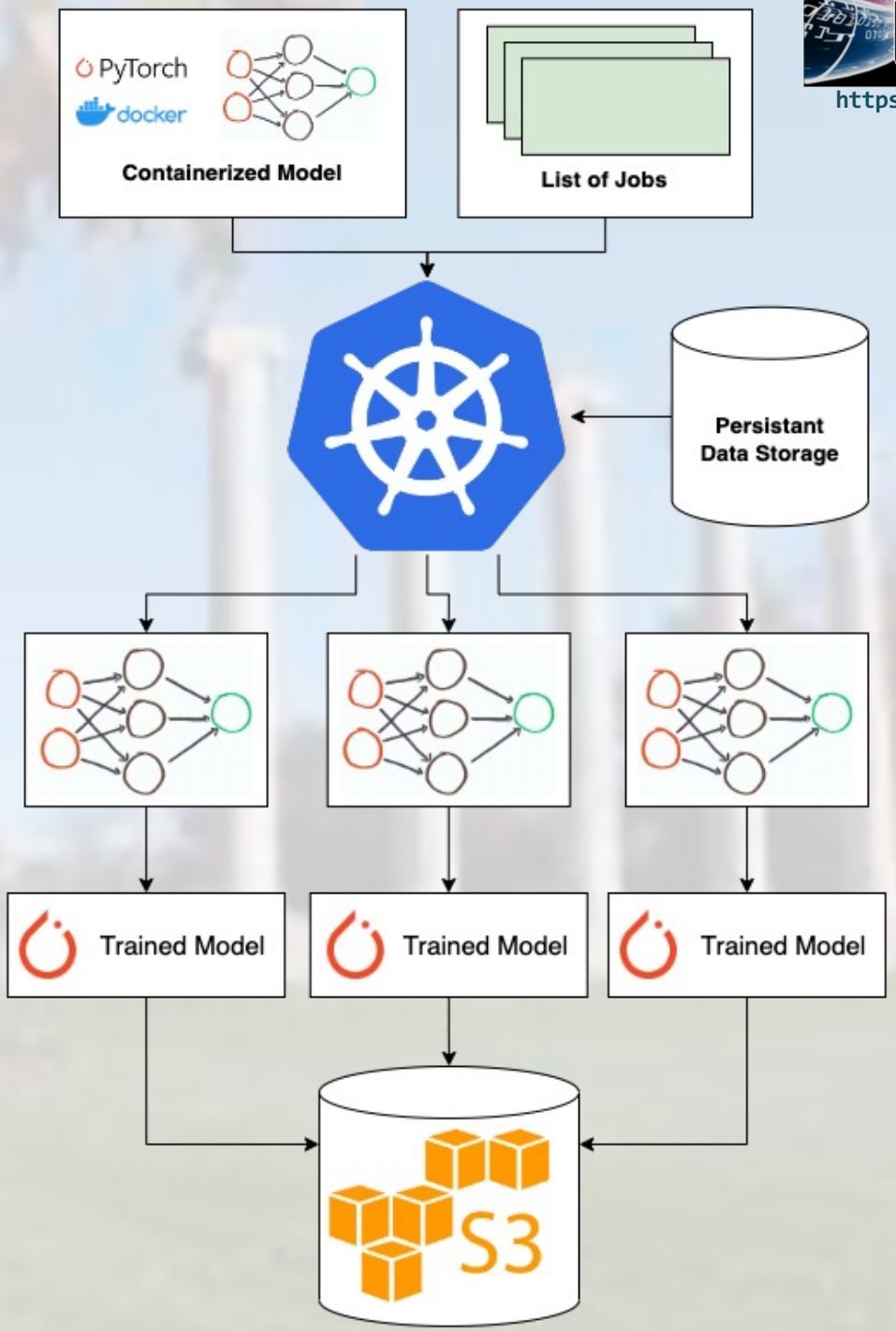
- Countless number of models available in modern CV
  - Model training time and resource requirements are oftentimes the bottleneck
- Models built for ground-photo CV do not always translate effectively to RSI applications
  - We want to train & evaluate state-of-the-art Deep Neural Networks (DNNs) on RSI datasets
- Several CV tasks in RSI Applications
  - Classification
  - Segmentation
  - *Object Detection*





System Overview:

Nautilus for Accelerated Deep Learning Research



Containerization of Deep Neural Architectures

- Deep Learning Libraries contain implementations of popular DNN Object Detection architectures
  - Faster R-CNN, YOLO, SWIN, etc.
- Availability of pretrained weights from large, CV datasets, which enables fine tuning on RSI datasets
  - MS COCO and ImageNet-1K
- Utilization of NVIDIA NGC as base container image
  - Optimized PyTorch containers for Deep Learning
- Containerized two PyTorch-based Object Detection Deep Learning Libraries:
  - MMDetection (Open MMLab)
  - Detectron2 (Facebook AI Research)

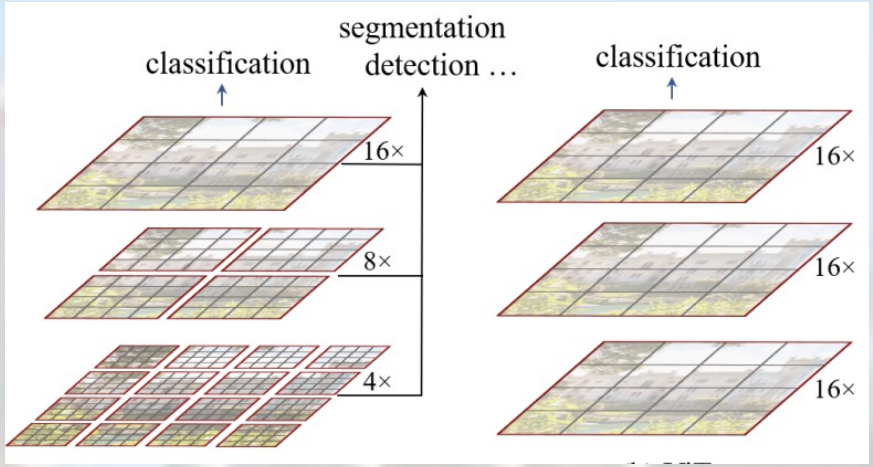
PyTorch | Accelerated with NVIDIA

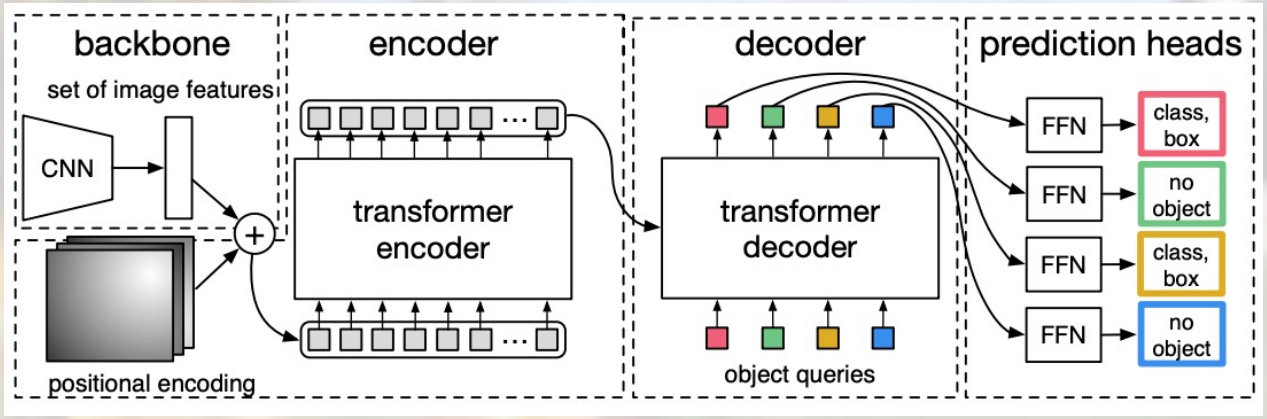
MMDetection

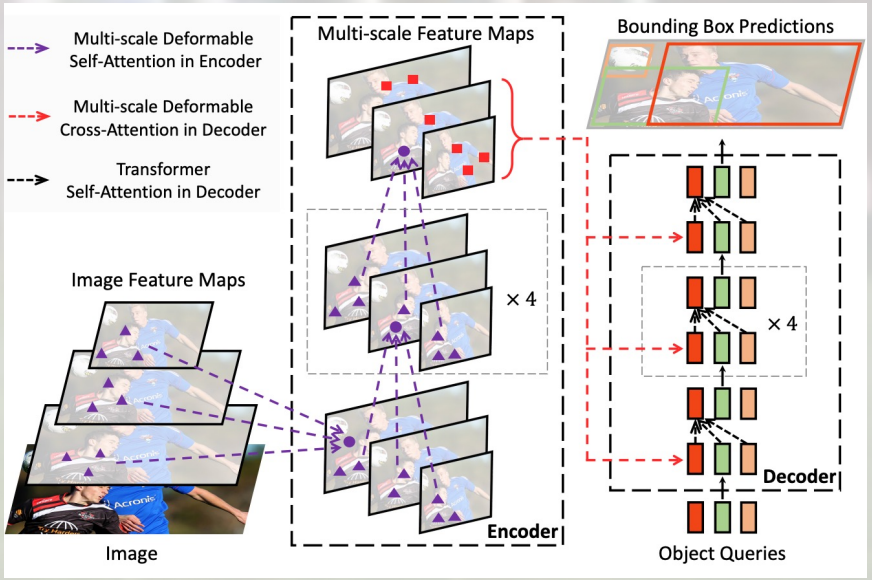
Detectron2

Models Containerized: Transformers

- SWIN Transformer (2021)
- DETR (2020)
- Deformable DETR (2020)
- ViT (2020)



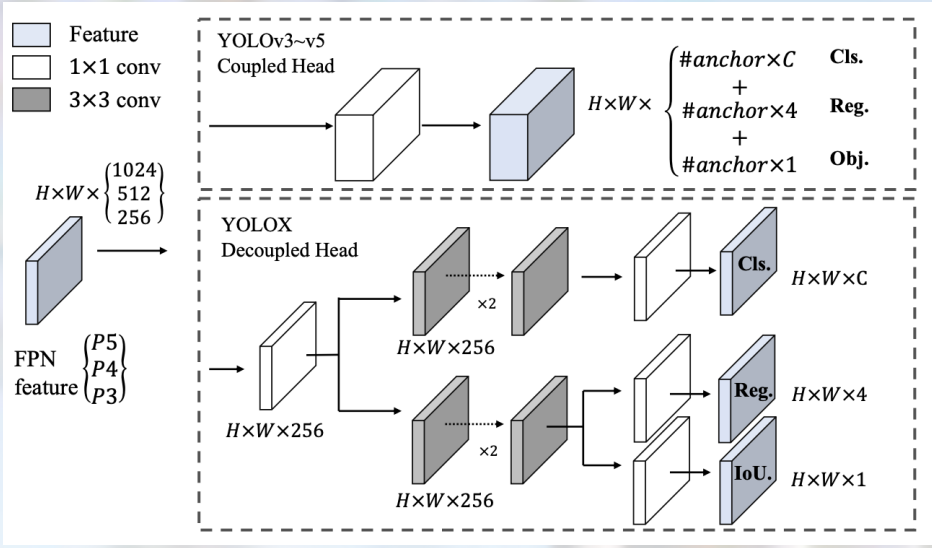


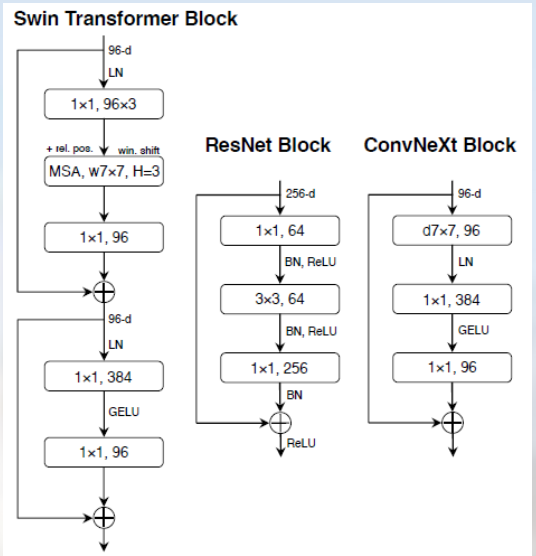


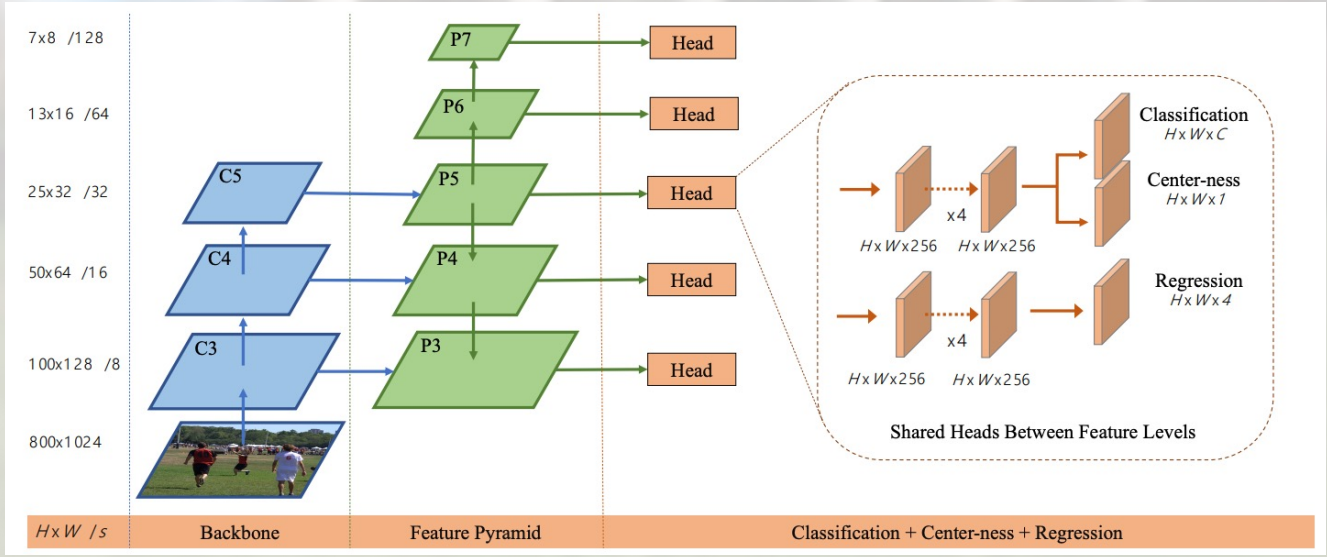
Images taken from original architecture publications

Models Containerized: Convolutional Networks

- ConvNext (2022)
- YOLOX (2021)
- FCOS (2020)
- YOLOv3 (2018)
- RetinaNet (2017)
- SSD (2016)



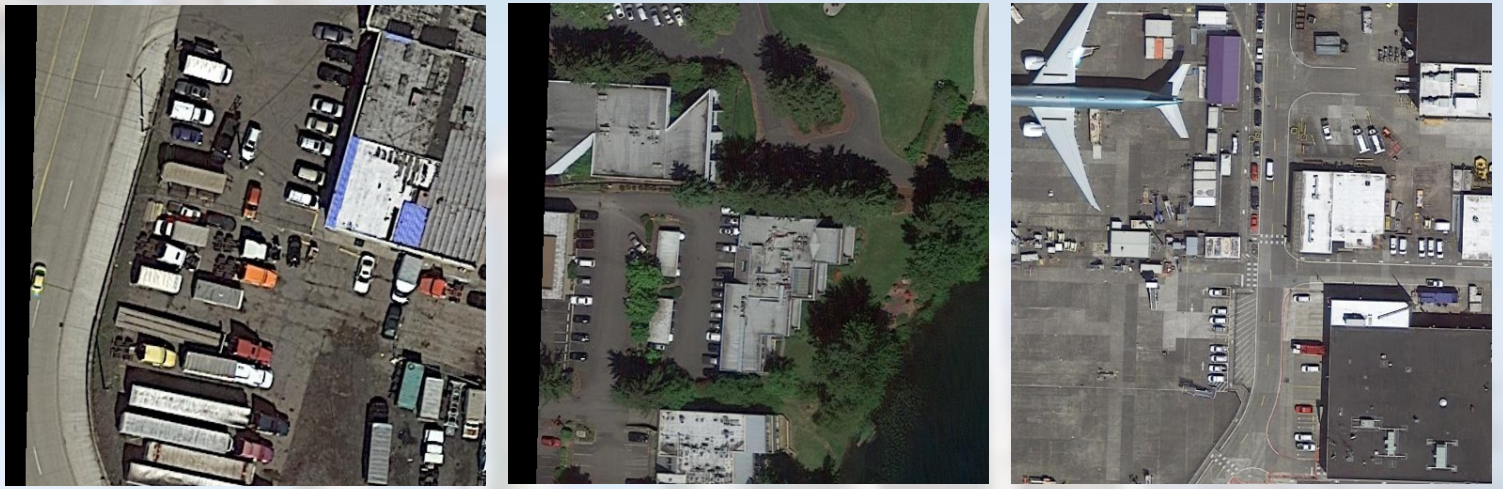




Images taken from original architecture publications

Detection Datasets

- DOTA (version 1.5)
  - Overhead Remote sensing dataset
  - 1869 scenes with 280,196 objects in 16 classes
  - Imagery chipped to enable better GPU acceleration
- RarePlanes (real data)
  - Pre-tiled, Pan-sharpened RGB dataset
  - Aircraft detection dataset
  - 17,050 tiles containing 25,205 aircraft belonging to 7 classes
    - 3 commercial aircraft classes
    - 4 military aircraft classes



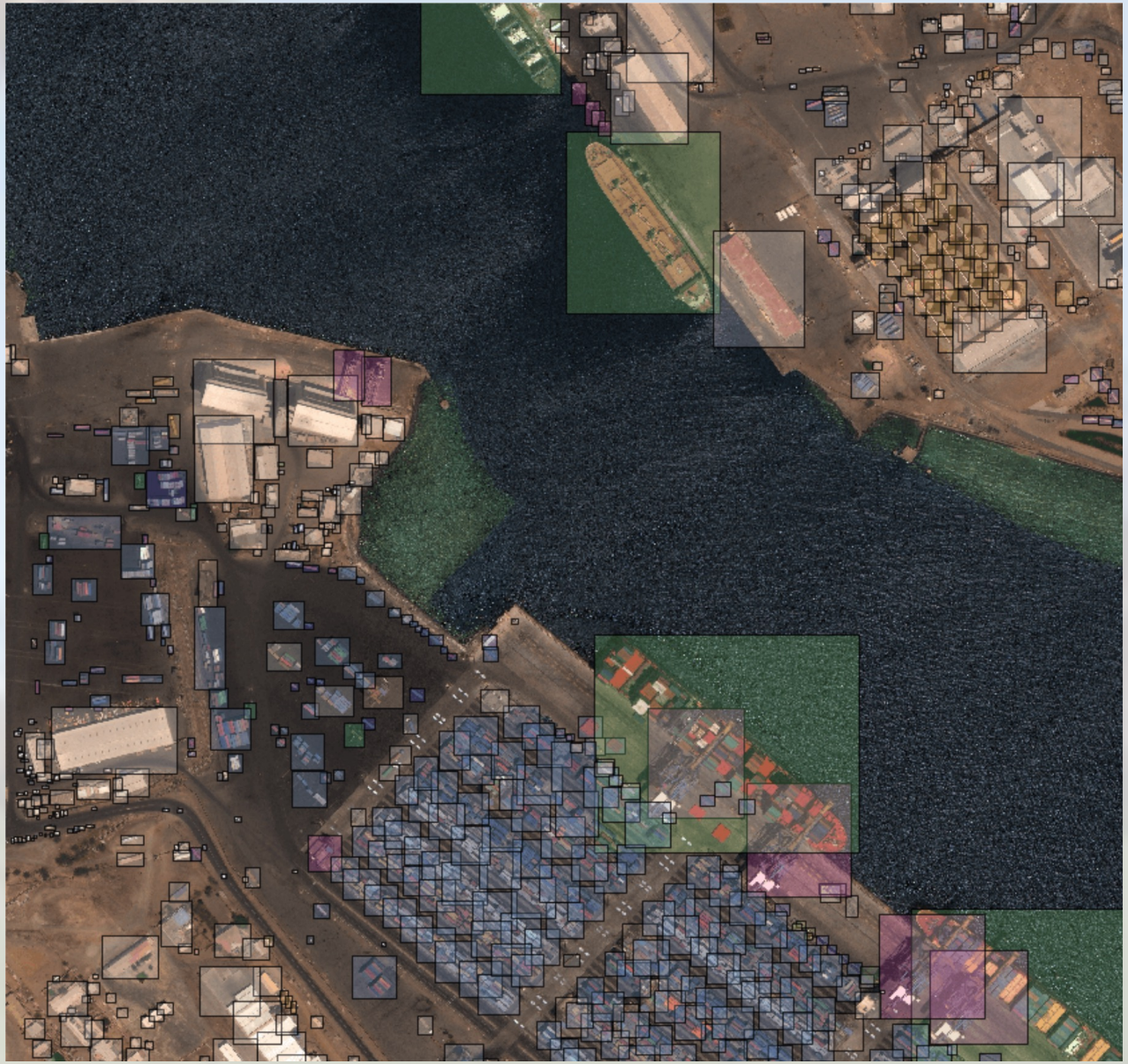
DOTA

RarePlanes

1. Xia, Gui-Song, et al. "DOTA: A large-scale dataset for object detection in aerial images." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.
2. Shermeyer, Jacob, et al. "Rareplanes: Synthetic data takes flight." *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2021.

Detection Datasets (cont.)

- XView
  - Released by DIUx in 2018
  - Over 1 million objects
  - 61 classes (60 + BG)
  - Heavily unbalanced
    - Most sparse class: 17 labels
    - Most dense class: 316,765 labels



Lam, Darius, et al. "XView: Objects in context in overhead imagery." *arXiv preprint arXiv:1802.07856* (2018).

Automating Kubernetes Jobs

- Creation of template Kube Spec YAML
- Bash scripting combined with environment variables to set the Dataset and Model to train
- Human time: ~3 hours

```
spec:
  template:
    spec:
      containers:
        - name: myContainer
          image: $CONTAINAER_IMAGE
          workingDir: $WORKDIR
```

Template YAML

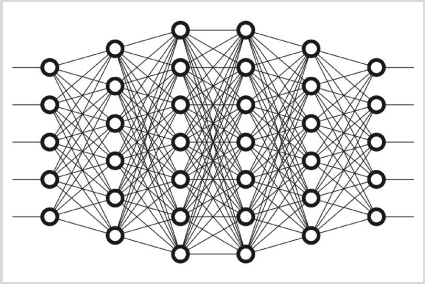
```
Dirs="mydir1 mydir2 mydir3 mydir4"
Container="ubuntu:20.04"

for Dirpath in $Dirs; do
    CONTAINER_IMAGE=$Container WORKDIR=$Dirpath envsubst < template.yml | kubectl apply -f -
done
```

Bash Script

Deep Learning on Nautilus: By the Numbers

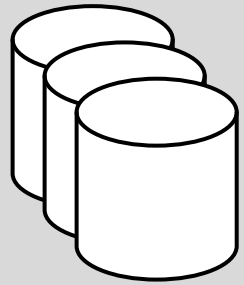
Nautilus



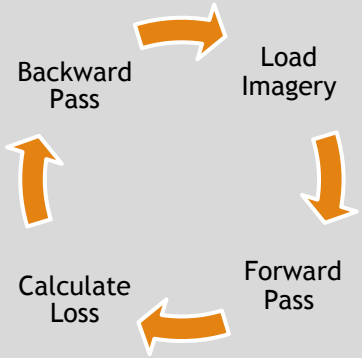
9 Deep Neural Architectures

27 Trained Models

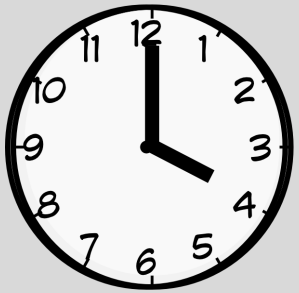
124.74 TB Imagery Processed



3 Open Source HR-RSI Datasets



- 8,100 Epochs
- 30M iterations
- 1.7B parameters



Wall Clock Time: 76 days, 10 hours

University of Missouri

https://www.freecodecamp.org/news/want-to-know-how-deep-learning-works-heres-a-quick-guide-for-everyone-1aedeca88076/
https://commons.wikimedia.org/wiki/File:PyTorch_logo_icon.svg
https://www.clipartmax.com/download/m2K9A0m2b1m2i8i8_4-o-clock-clip-art-clock-face-clip-art/

Future Work: Kubernetes and S3 Python APIs

- Both AWS and Kubernetes publish Python packages with APIs to perform tasks in Python
- Move from ad-hoc bash scripting to Python application that can better manage the creation of training jobs
- Combine K8s and AWS Python API to automate copying of trained models and performance results to S3 bucket