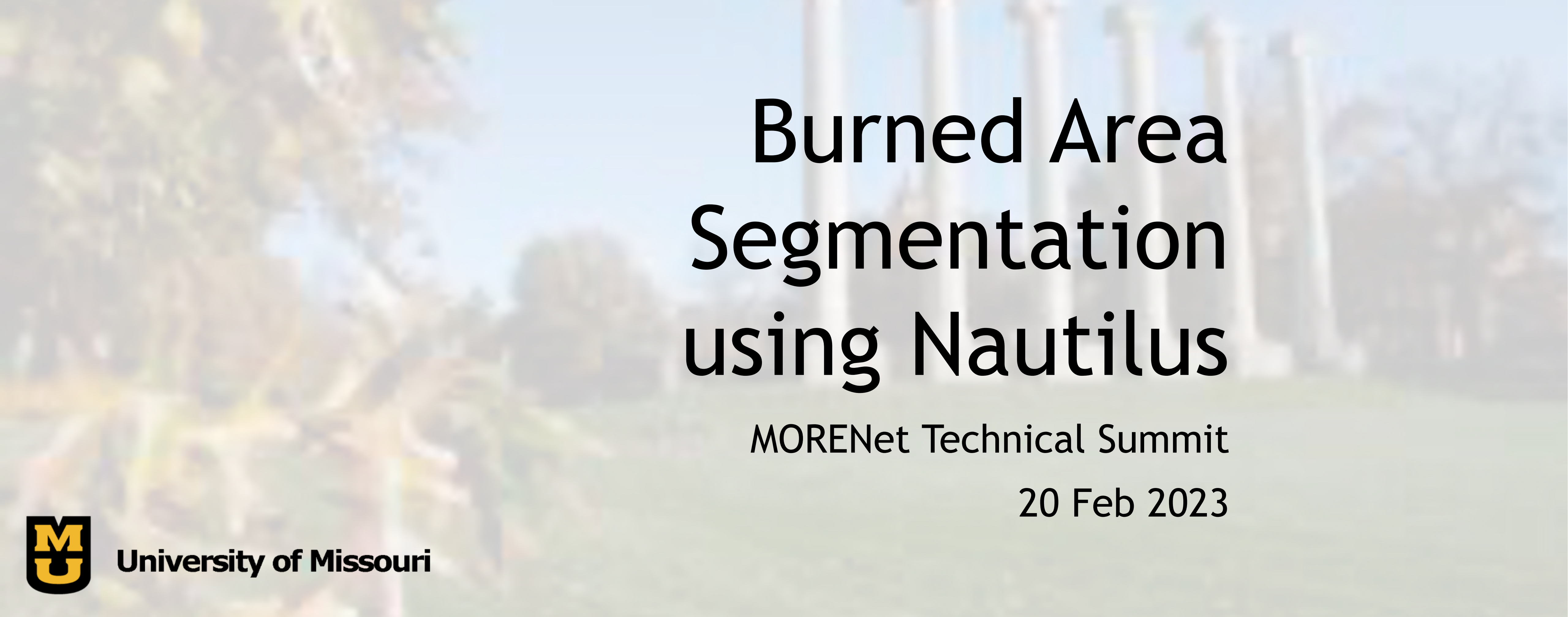
High-Performance Data-Intensive Computing Systems Laboratory



Burned Area Segmentation using Nautilus

MORENet Technical Summit

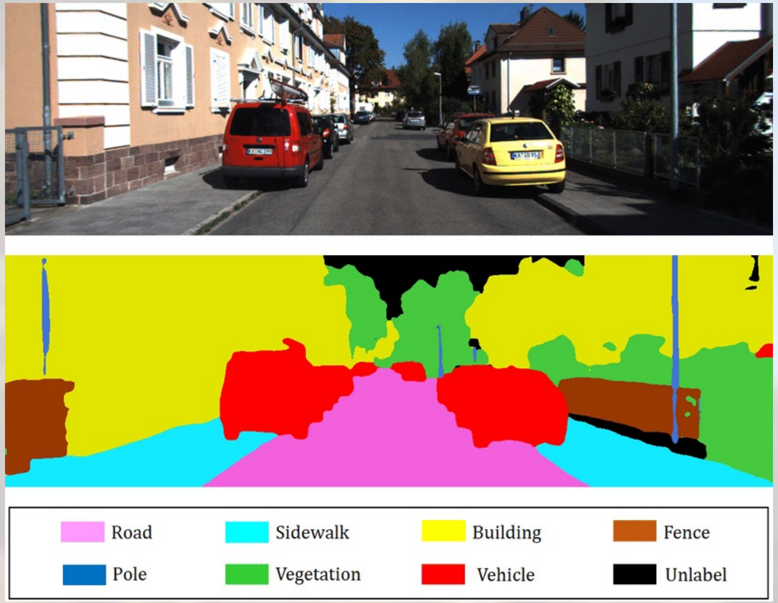
20 Feb 2023

University of Missouri

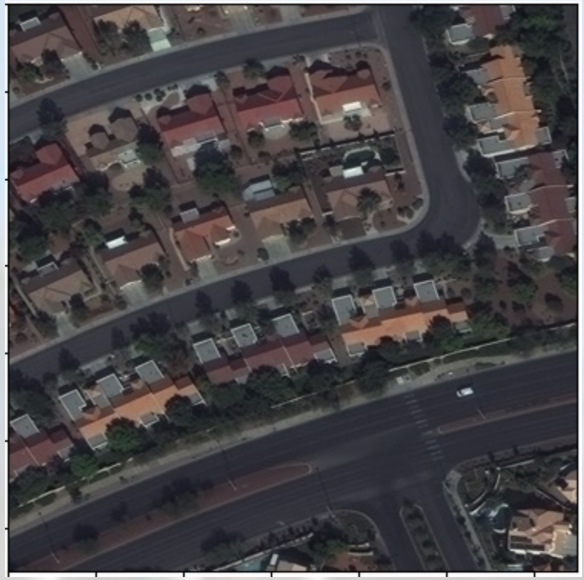
Outlines

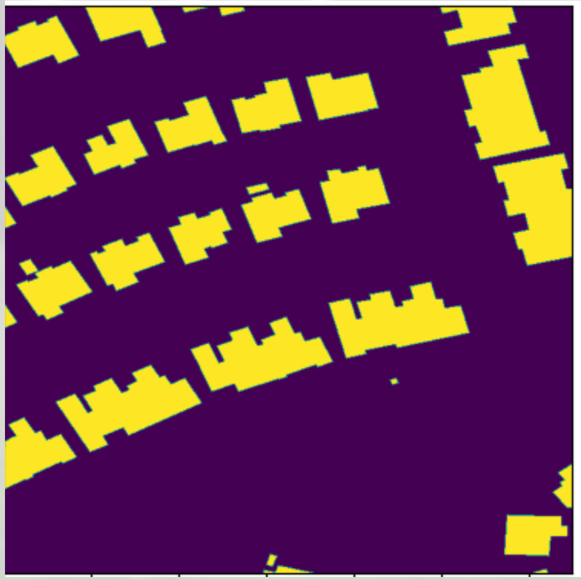
1. What is the problem
2. What is the data
3. What is the tool
4. Preparing workspace
5. Generating yml files
6. Training cycle sample
7. Results sample
8. Statistics about using Nautilus

What is the problem



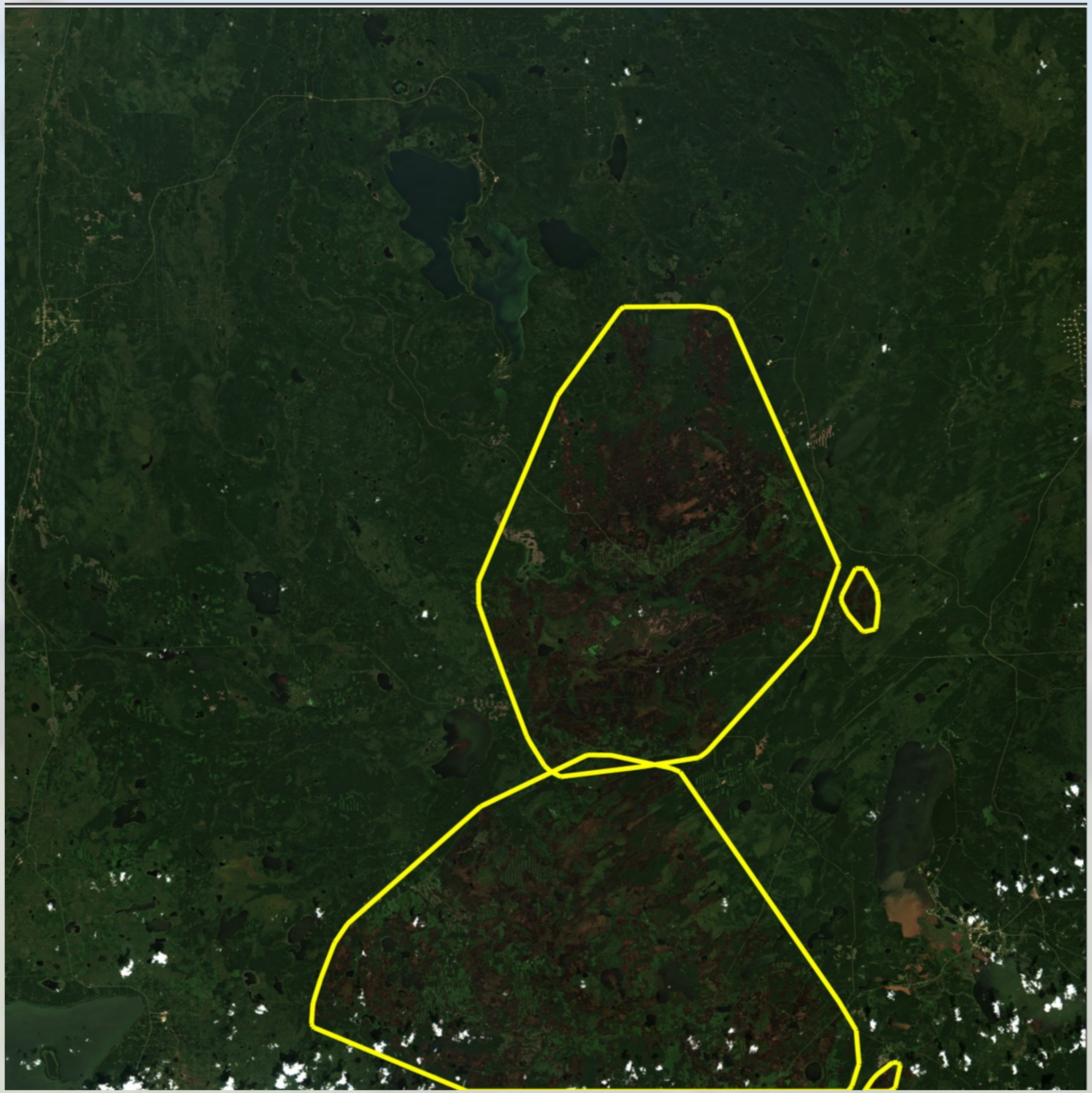
Semantic Segmentation:
The precise classification of every pixel in the image into one of the classes





What is the data

- Hyperspectral satellite imagery
- Geotagged
  - It has a coordinate reference system
  - Each pixel is also a coordinate on the earth
- It has multiple bands
  - Visible range (R,G,B)
  - NIR
  - SWIR



What is the tool

Convolutional Neural Network (CNN)

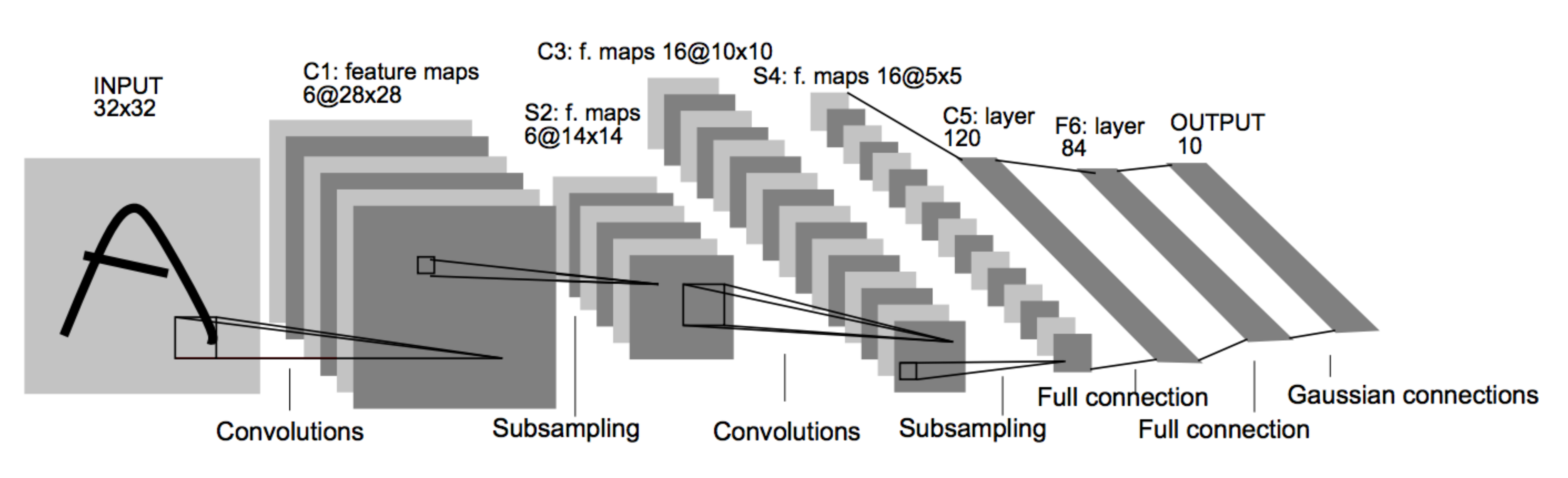
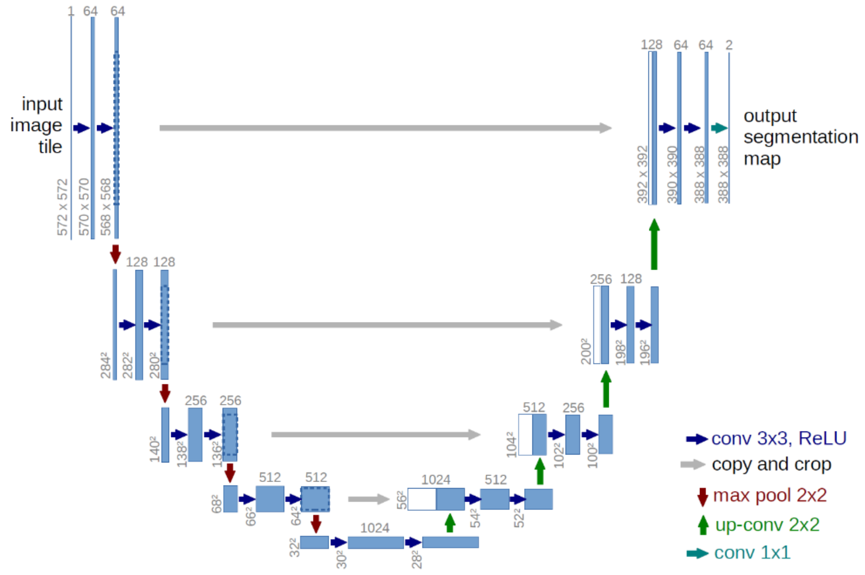


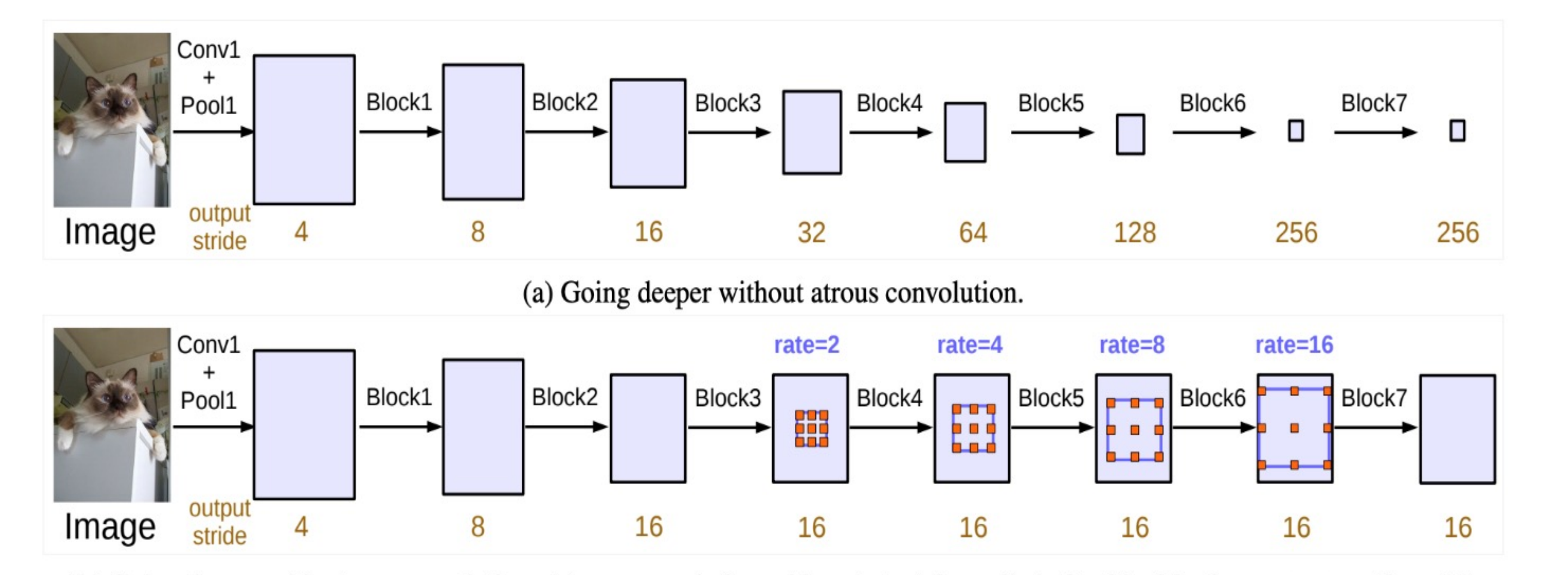
image classification

Target detection

Semantic segmentation

What is the tool





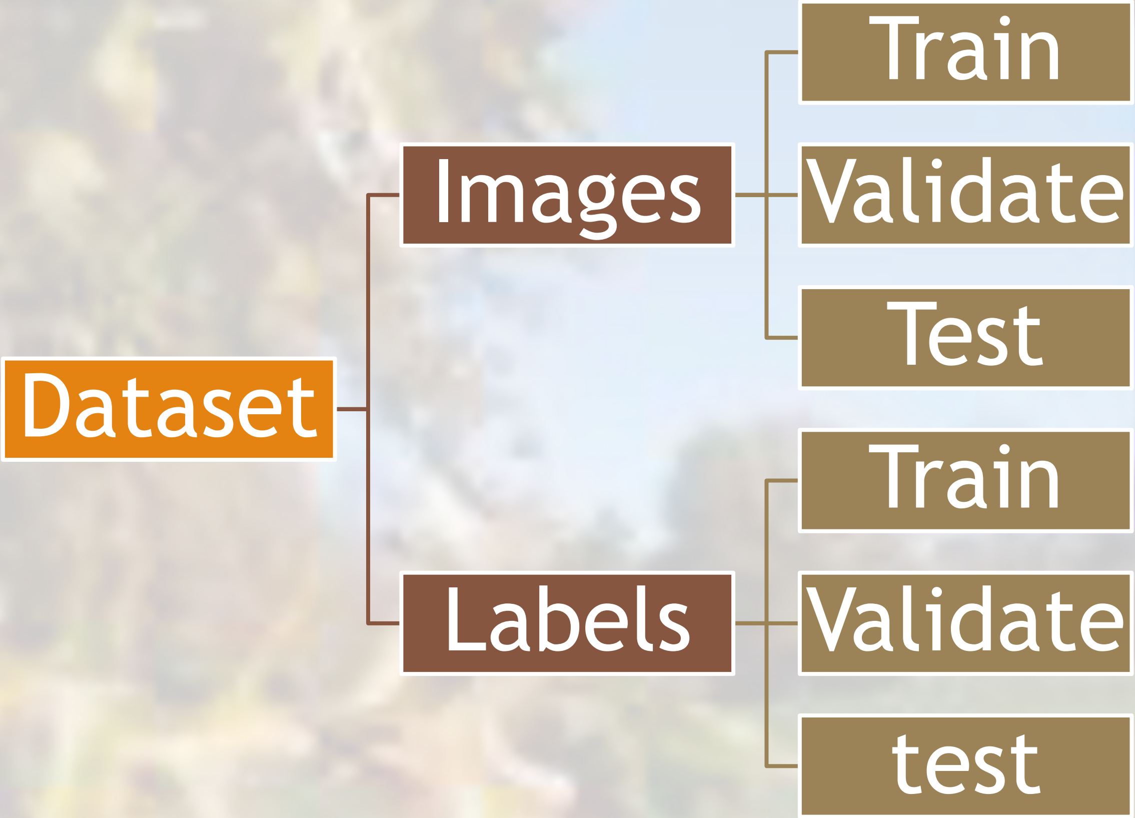
(a) Going deeper without atrous convolution.

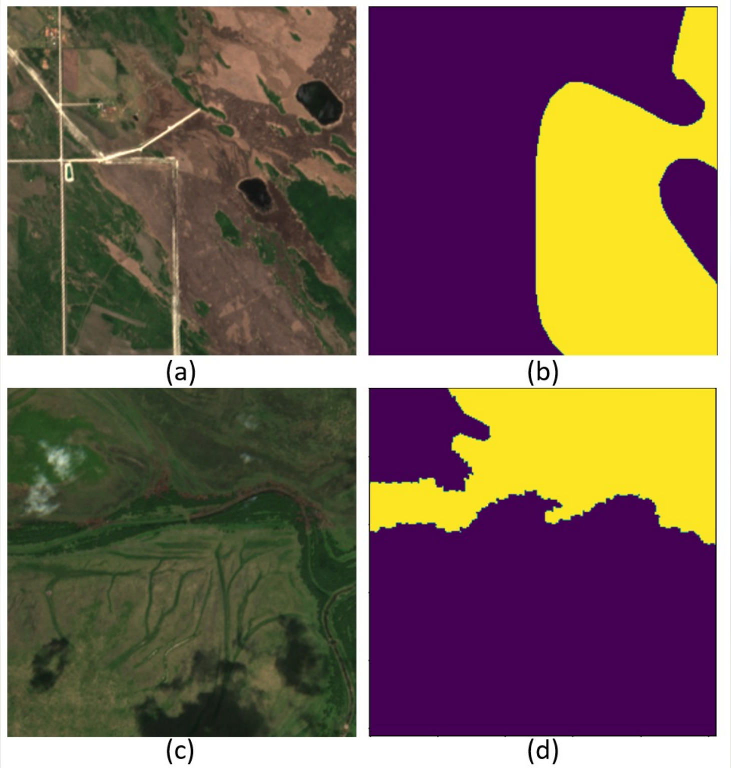
U-Net

Deeplab

Preparing workspace

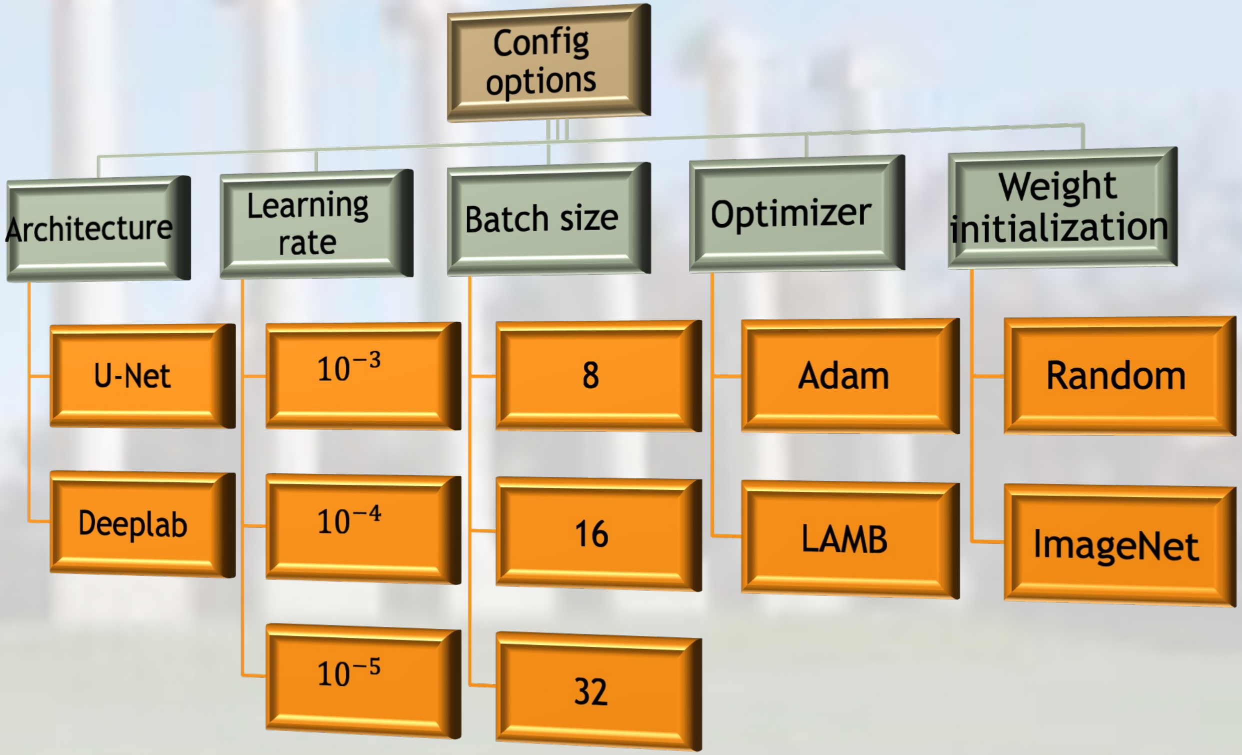
Dataset




(a) (b) (c) (d)

Preparing workspace
config files

- The config file allows us to specify options for each model
  - The architecture to be used
  - The learning rate
  - The batch size
  - The optimizer



Preparing workspace
config files

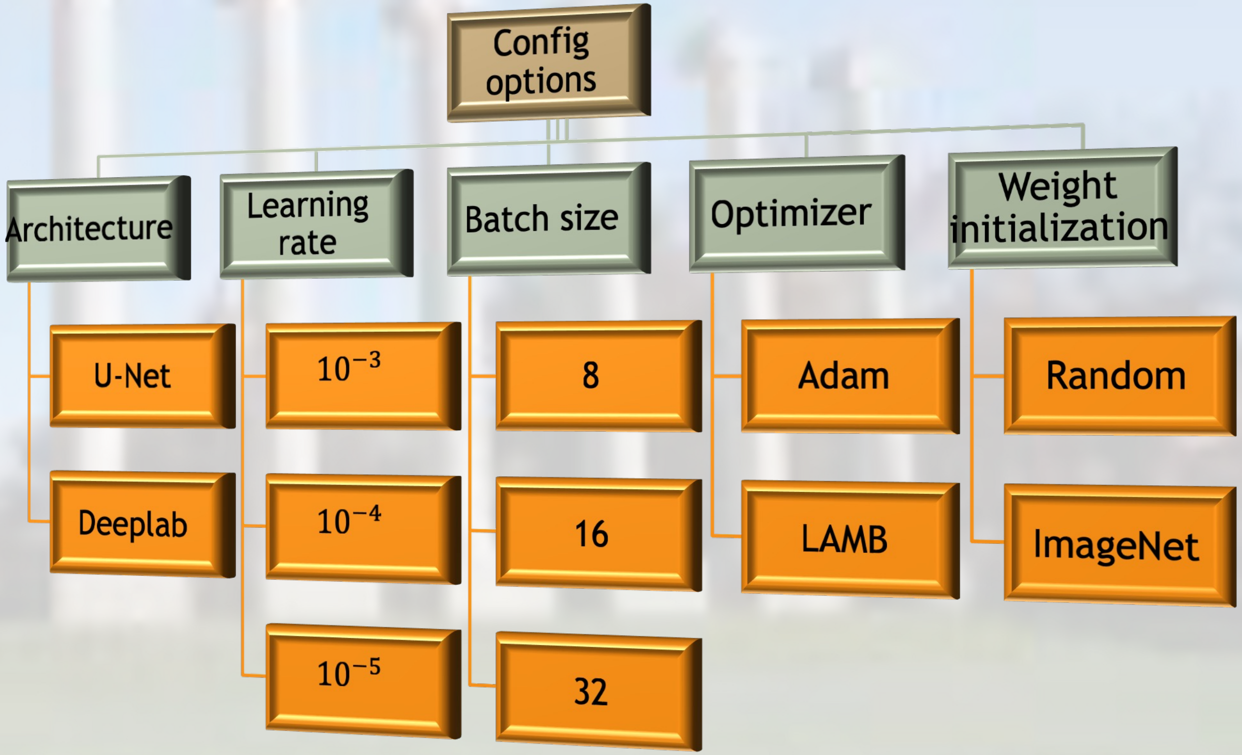
```
{
        "epochs": 100,
        "model_architecture": "Unet",
        "model_args": {
                "encoder_name": "resnet50",
                "encoder_weights": "imagenet"
        },
        "gpus": [0],
        "batch_size_per_gpu": 4,
        "num_classes": 2,
        "training_data": [{
                "image_dir": "/canada2019-3/training_data_png/tcis/train",
                "mask_dir": "/canada2019-3/training_data_png/labs/train"
        }],
        "optimizer_name": "Adam",
        "optimizer_args": {
                "lr": 0.001
        },
        "validation_data": [{
                "image_dir": "/canada2019-3/training_data_png/tcis/validate",
                "mask_dir": "/canada2019-3/training_data_png/labs/validate"
        }],
        "validate_while_training": true
}
```

Preparing workspace
yaml files

- Each combination of these parameters provides us with a unique conditions for an experiment to run

- $Number\ of\ experiments = 2 \times 3 \times 3 \times 2 \times 2$
  $Number\ of\ experiments = 72$

- For each experiments we save two models:
  - The last model
  - The best model
  - We end up with 144 models



Config options

Architecture
- U-Net
- Deeplab

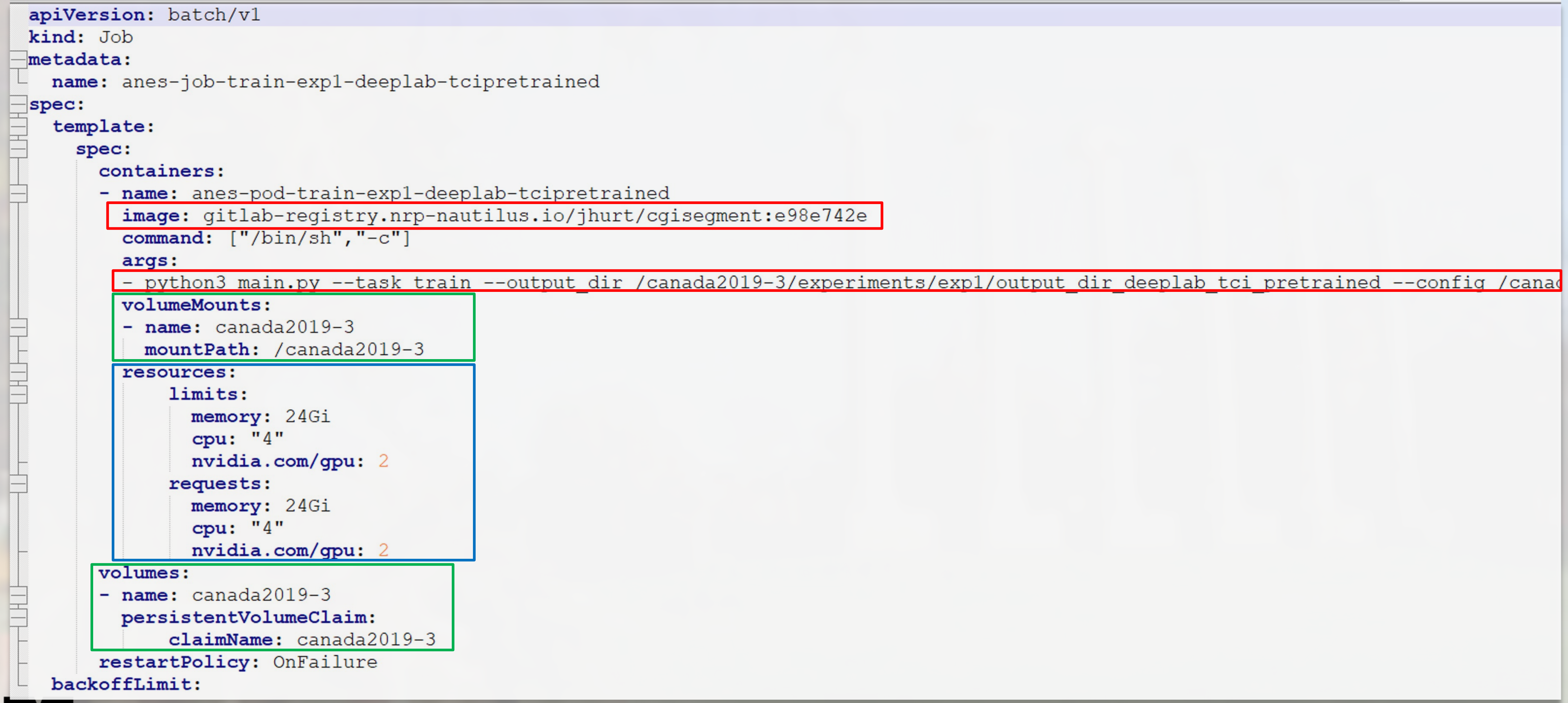
Learning rate
- 10^{-3}
- 10^{-4}
- 10^{-5}

Batch size
- 8
- 16
- 32

Optimizer
- Adam
- LAMB

Weight initialization
- Random
- ImageNet

Preparing workspace
yaml files



```yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: anes-job-train-exp1-deeplab-tcipretrained
spec:
  template:
    spec:
      containers:
      - name: anes-pod-train-exp1-deeplab-tcipretrained
        image: gitlab-registry.nrp-nautilus.io/jhurt/cgisegment:e98e742e
        command: ["/bin/sh","-c"]
        args:
        - python3 main.py --task train --output_dir /canada2019-3/experiments/exp1/output_dir_deeplab_tci_pretrained --config /canad
        volumeMounts:
        - name: canada2019-3
          mountPath: /canada2019-3
        resources:
            limits:
              memory: 24Gi
              cpu: "4"
              nvidia.com/gpu: 2
            requests:
              memory: 24Gi
              cpu: "4"
              nvidia.com/gpu: 2
      volumes:
      - name: canada2019-3
        persistentVolumeClaim:
            claimName: canada2019-3
      restartPolicy: OnFailure
  backoffLimit:
```

Preparing workspace

Auto mass generating of config/yml files

- Config files were autogenerated
- Yml files were autogenerated

Using Jupyter notebook
Jinja library

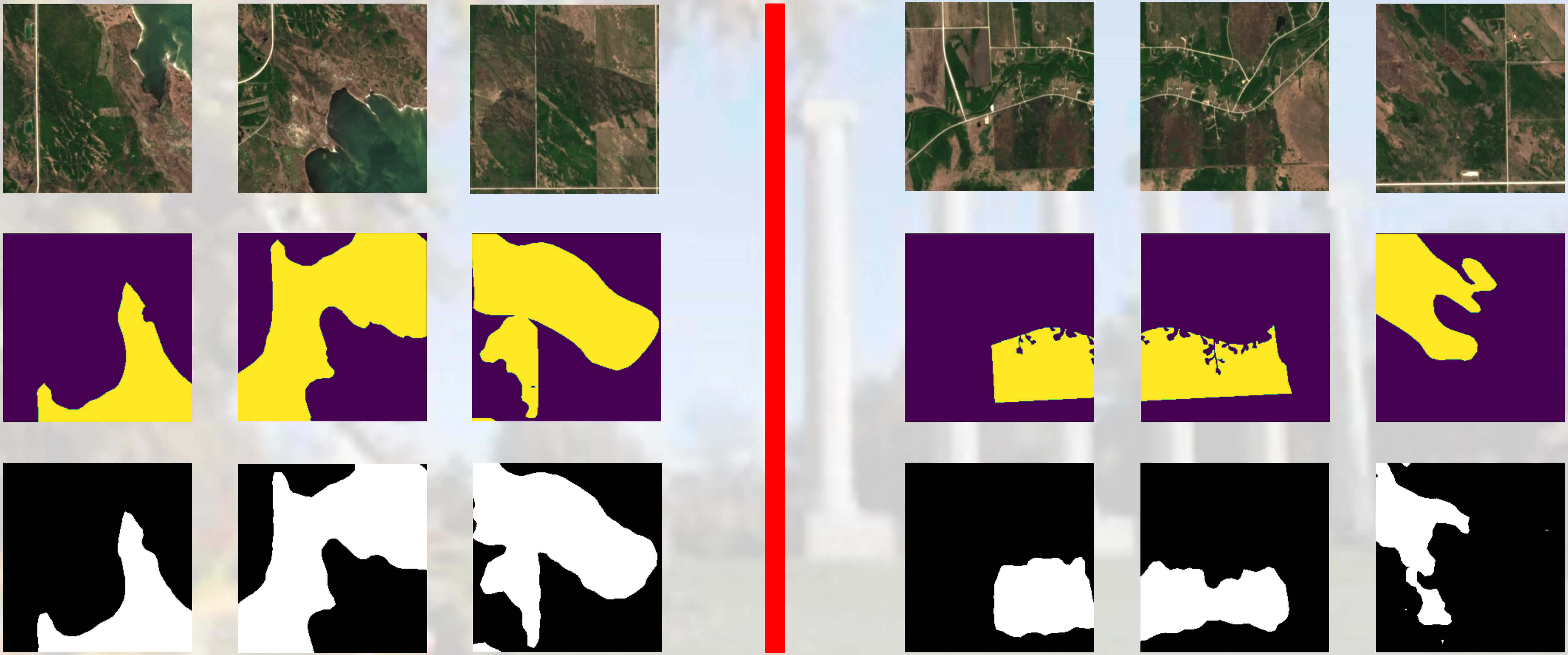
- Jobs were auto submitted
- Jobs were auto deleted after completion

Using Windows/Linux bash files

Sample Automation Script

```
@ECHO OFF

Rem This batch file executes kubectl commands to create training jobs

::echo %kubectl%
SET exp_list=2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

(for %%a in  (%exp_list%) do (
      echo %%a

      kubectl create -f experiments\exp%%a/job_exp%%a_deeplab_img.yaml
      kubectl create -f experiments\exp%%a/job_exp%%a_deeplab_tci.yaml
      kubectl create -f experiments\exp%%a/job_exp%%a_deeplab_img_pretrained.yaml
      kubectl create -f experiments\exp%%a/job_exp%%a_deeplab_tci_pretrained.yaml
))

echo "batch complete"%
```

Results sample



Statistics about using Nautilus

- Iterations of Training Completed: 515,550
- Number of images Processed: 7,070,400
- Trainable Parameters Optimized: 23 millions per model
- The time it took to prepare the experimental set up and to run all the training sessions in parallel is 12 hours
- The actual time it would have take to train is 21 days 12 hours 45 minutes