# Rail Network Detection From Aerial Imagery Using Deep Learning

Mehrdad Salehi, Yonghong Wang

## Introduction

- The goal of this project is to classify and segment the rail network from aerial imagery using Deep Learning.
- Traditionally, creating such a network needs manual digitization of satellite imagery and building the network with human intervention.
- Per our knowledge, it is the first time to use deep learning to detect and segment rail network from satellite imagery data.

## Applications

- To build and maintain the rail network on the map:
  - Map is outdated and does not show a new rail network, but a recent aerial imagery shows a new network was built there
  - Map shows a rail network, but in reality such network does not exist anymore.

## Challenges

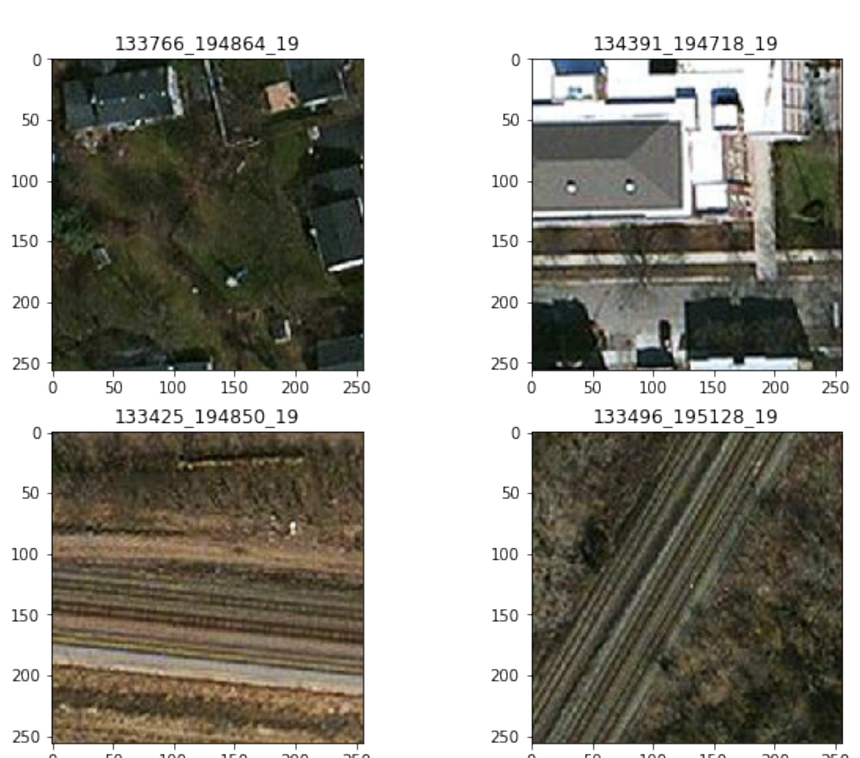
- Obtaining a clean label dataset
  - Label data and aerial imagery not temporally in sync
  - Label data is shifted, missing, inaccurate, etc.
- Rail network is not always visible in the aerial imagery:
  - Objects (e.g., trees, shadows, buildings) could obstruct rail network
  - Rail network is not visible in low-resolution aerial imagery

## Related Work

Mnih et al., in [1] proposed a large-scale learning approach to detect roads using a neural network. They predict a small block of pixels is road or not. With post processing, they got about 87% test accuracy.

## Data

- Satellite imagery tiles. Each tile covers around 60x60 meters (zoom 19 tiles)
- Open Street Map as a source to build label data




## Classification

### Labels

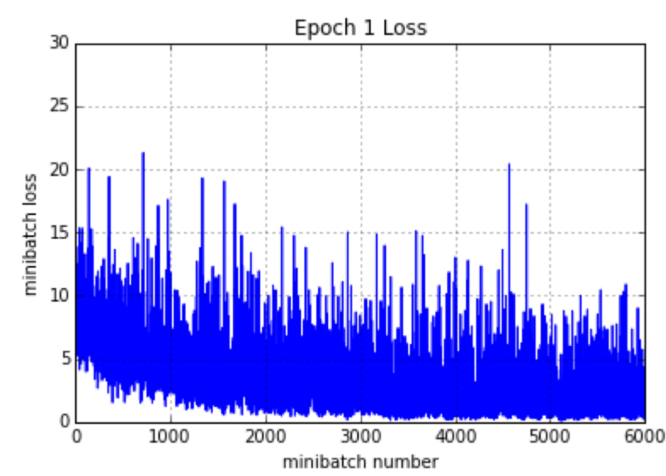
Applied large-scale geometric data processing using Spark, Scala, and JTS to:

- Intersect the Open Street Map rail network data with imagery tiles
- Label a tile whether it contains rail tracks or not, using JTS to calculate the geometric relationship between the tile and rail geometries
- Label data was created for entire US

### Model

| Layer | Size |
|---|---|
| Conv1 | 3 X 3 X 96 |
| Max-pool | 2 X 2 |
| Conv2 | 3 X 3 X 96 |
| Max-pool | 2 X 2 |
| Conv3 | 3 X 3 X 96 |
| Max-pool | 2 X 2 |
| Conv4 | 3 X 3 X 96 |
| Max-pool | 2 X 2 |
| Conv5 | 3 X 3 X 96 |
| Max-pool | 2 X 2 |
| Conv6 | 3 X 3 X 96 |
| Max-pool | 2 X 2 |
| Conv7 | 3 X 3 X 96 |
| FC1 | 128 X 128 |
| FC2 | 128 X 64 |
| FC3 | 64 X 2 |

### Results




| Training Size | Epochs | Accuracy |
|---|---|---|
| 10K | 1 | 84% |
| 100K | 1 | 96% |
| 100K | 5 | 98% |

### Test Result Analysis

Map Label: rail, Pred: rail

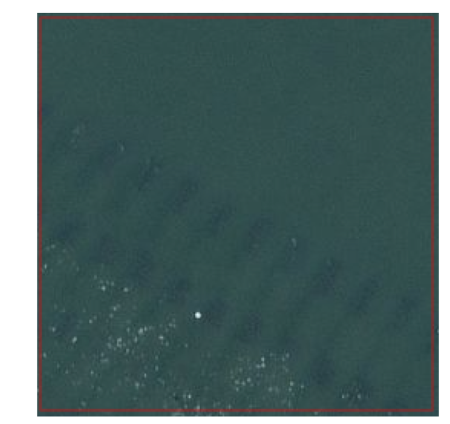
Map Label: non-rail, Pred: non-rail

Map Label: rail, Pred: non-rail

Tree Shadows can be the reason of misclassification

Rail track is covered by grass, not used anymore

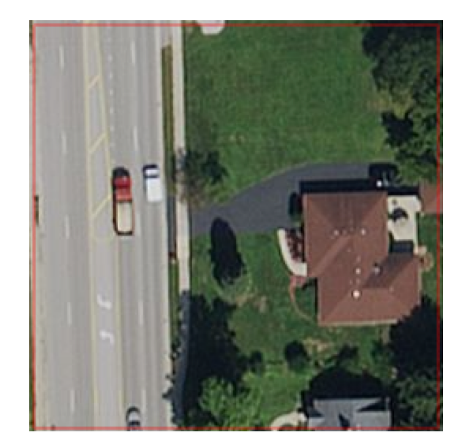
Floating object?



Map Label: no-rail, Pred: rail

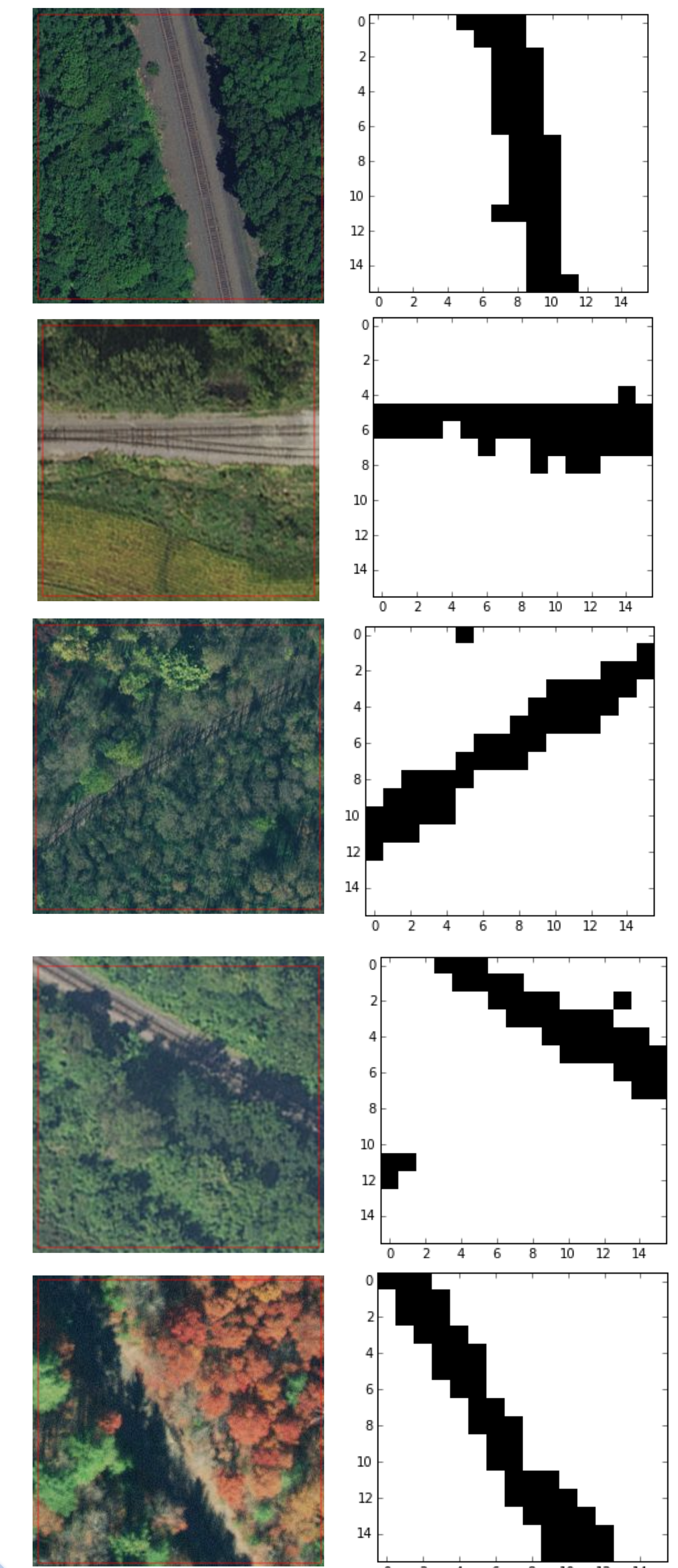
The pattern of bridge is very similar to a rail track

The pattern of the sidewalk is similar to a rail track



Unused/Under construction

Segmentation: True positives



## Segmentation (zoom 23)

- The goal is to predict the labels for each zoom 23 tile
- We used a CNN model and tuned hyper-parameters of the model multiple times

### Labels

- Used large scale geometric data processing (Spark & JTS
- Intersected 3x3 meter tiles (zoom 23) with the OSM rail network data
- Generated labels for entire US

### Model

| Layer | Size |
|---|---|
| Conv1 | 7 X 7 X 96 |
| Max-pool | 2 X 2 |
| Conv2 | 5 X 5 X 128 |
| Max-pool | 2 X 2 |
| Conv3 | 3 X 3 X 128 |
| Conv4 | 3 X 3 X 256 |
| Conv5 | 3 X 3 X 128 |
| Conv 6 | 3 X 3 X 64 |
| Max-pool | 2 X 2 |
| Conv7 | 1 X 1 X 2 |

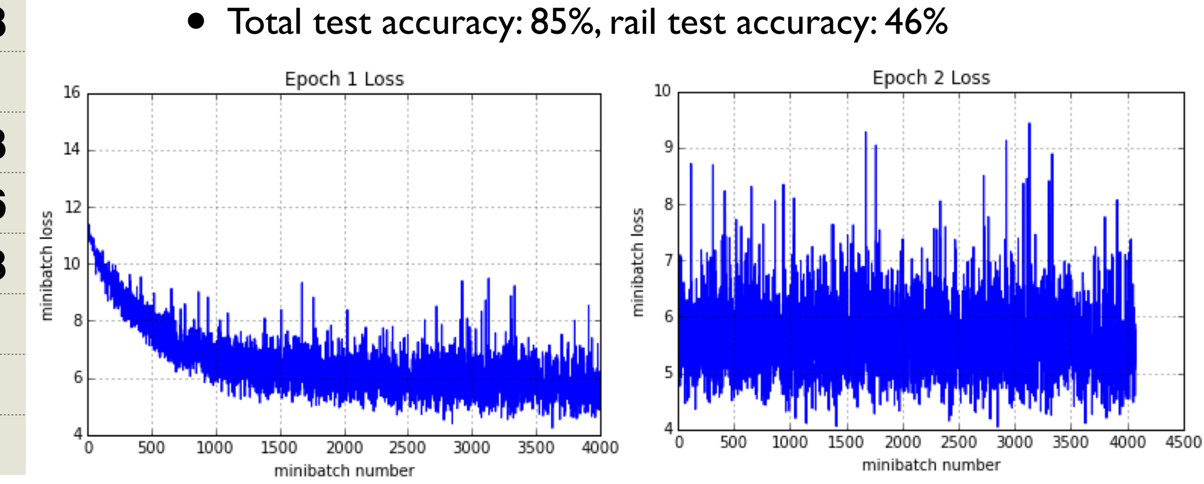
### Result

Training data was imbalanced in the following models:

- 200k training images, decay rate 0.96, decay steps 1000, epochs 2: Model barely learned
- 200k training images, decay rate 0.1, decay steps 1000, epochs 1: model barely learned

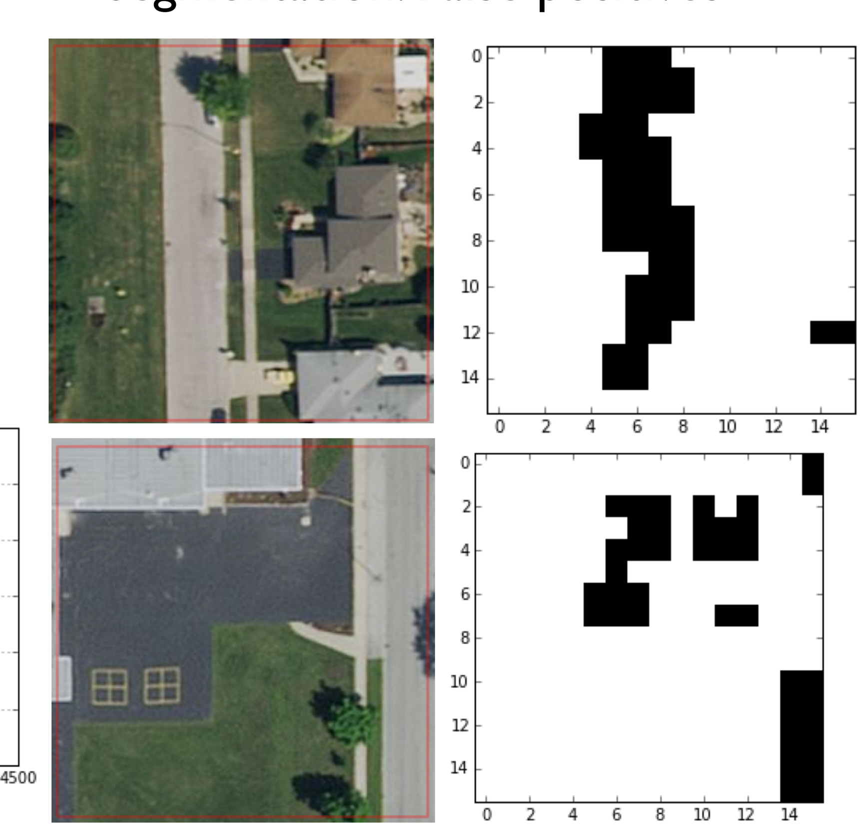
With balanced training data:

- 78k training images, decay rate 0.96, decay steps 1000, epochs 2: model learned much better.
- Total test accuracy: 85%, rail test accuracy: 46%




### Result Analysis

Segmentation: False positives



## Future Work

- In final report, we are going to try deconv network, and resNet for segmentation at zoom 23 level
- We will also try to do segmentation at pixel level (zoom 27). We will apply CNN and DeConv network, and smoothing method

## Literature

[1] V. Mnih and G. Hinton. Learning to detect roads in high-resolution aerial images, 2010.