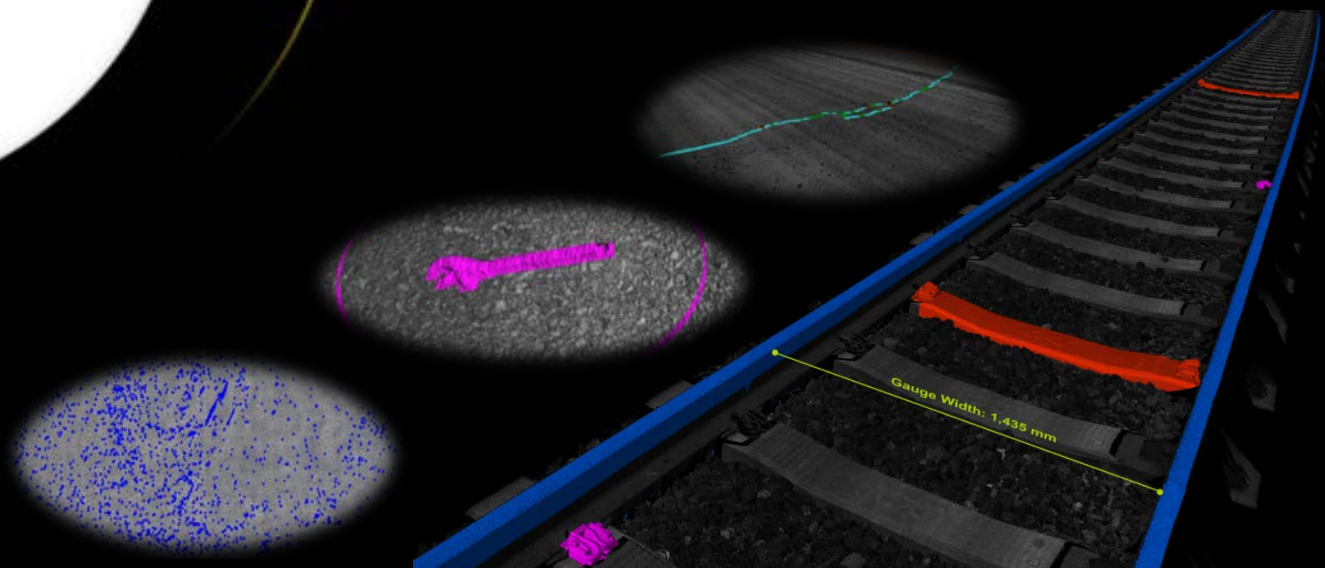
# Use of Deep Neural Networks for Railroad Inspection

Railmetrics

**Vision Technology for Inspection of Transportation Infrastructures**

www.railmetrics.com



# Researchers

**John Laurent**

**Masters of Electrical Engineering**

**Mario Talbot**

**Masters of Electrical Engineering**

# Rail/Pavemetrics: Sensors for Infrastructure Inspection

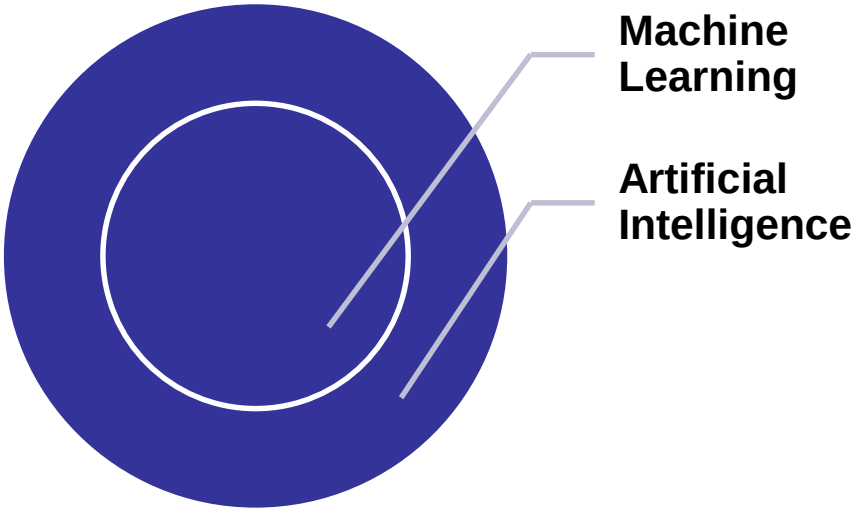

- **Founded 2009; a "Spin-off" of Canada's National Optics Institute (INO)**
- **3D sensor development**
- **Automated data processing algorithm development**
- **Technology used for Roads, Runways, Rails, Tunnels**
- **40+ countries**

# Machine Learning

**Machine Learning:**

**AI uses human-developed algorithms/logic to parse data, learn from data, then apply that knowledge to make subsequent decisions…**

**…used to solve problems humans already know how to solve**

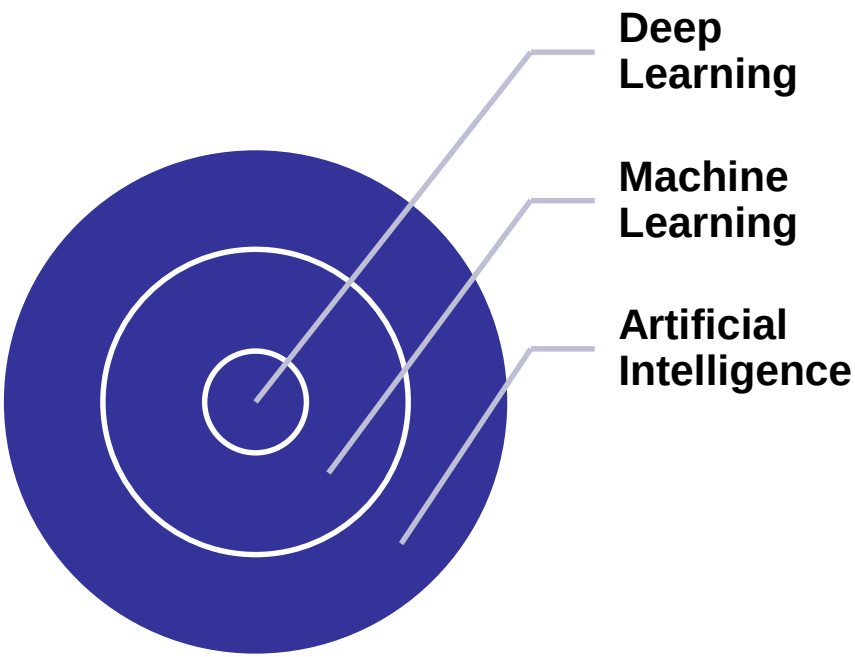


# The Advantage of Deep Learning

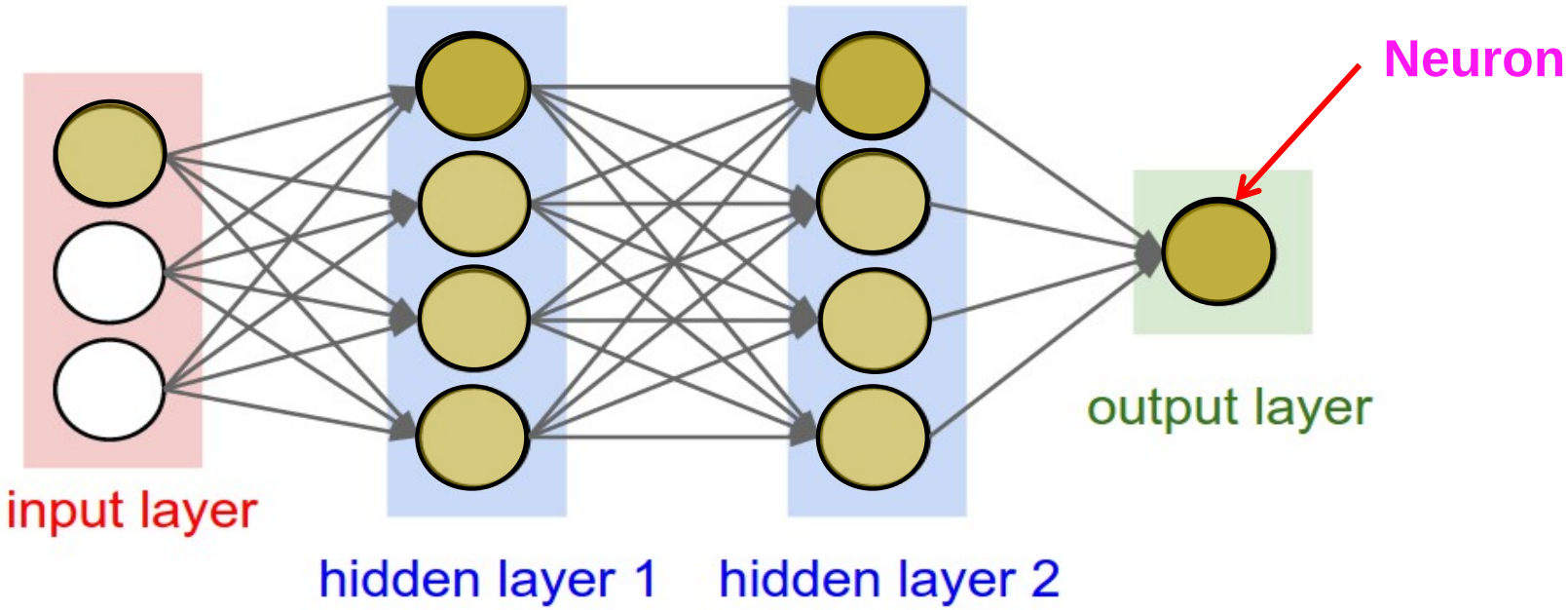
**Deep Learning:**

**AI *develops its own methods of analysis*, much in the same way that a human brain does, by using an Artificial Neural Network (ANN or DNN)…**

**…has the potential to solve new problems in new ways and does not require re-training as new examples are added**



# Anatomy of an Deep Neural Network (DNN)




(Stanford University, 2017)

("Deep" NN have at least 2 hidden layers)

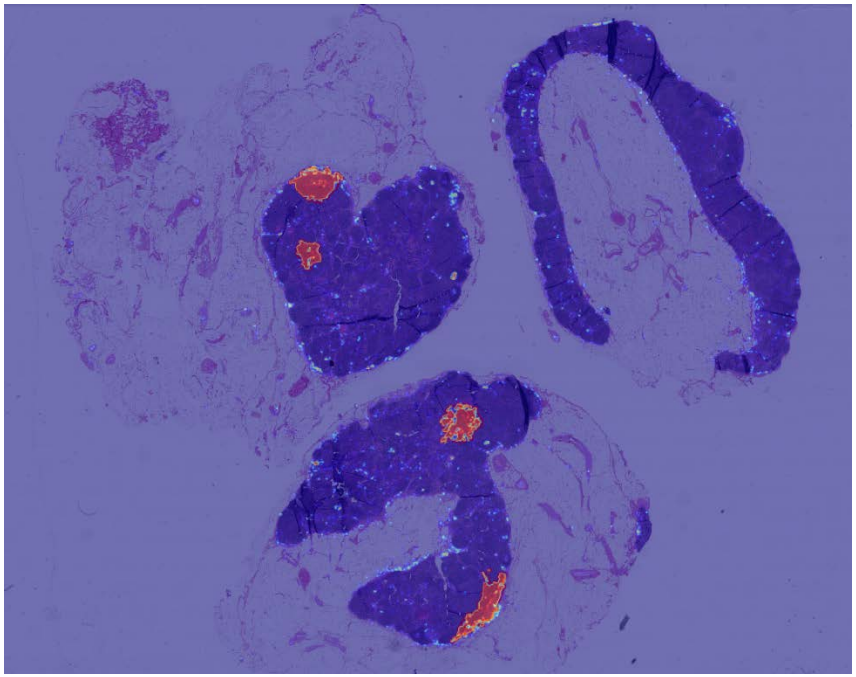
- **Made up of: Layers, Neurons and Weights**
- **In this example:**
  - **32 weights (each path through is 1 weight)**
  - **9 biases (neurons in hidden + output)**
  - **41 learnable parameters (weights + biases)**

# Training and Testing a Neural Network

- **Present data to AI + give it the correct answer**
- **Then test its ability to obtain correct answer for new data**
- **Repeat process until no further gains are made ("convergence")**
- **One full cycle through data is called an "Epoch"**
- **DNN error (difference between expected and actual classification) is calculated for each Epoch in order to adjust parameter weighting**
- **It usually requires at least 50 Epochs to train an AI**

# A DNN Case Study: 2016 Camelyon Grand Challenge

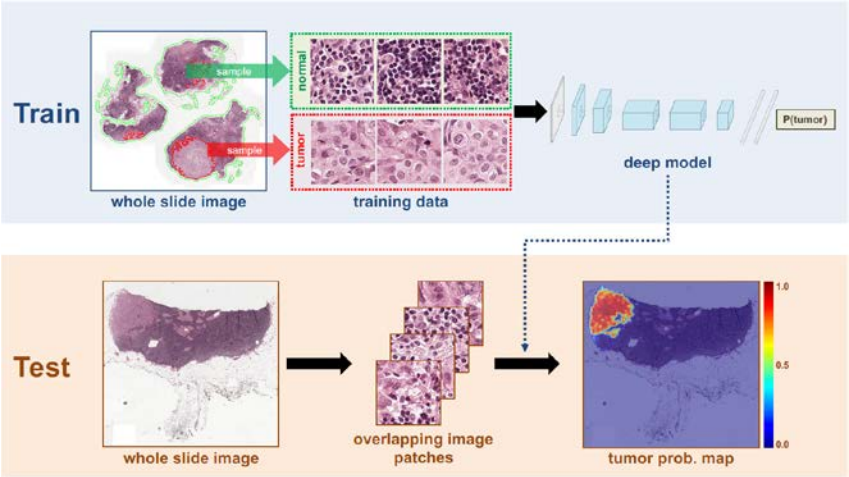
- **Manual review of images to detect cancer is time consuming**
- **Has a high cognitive load**
- **Suffers from a lack of standardization**
- **Results in diagnostic errors**
- **Can AI *assist* oncologists in this task?**



Deep Learning for Identifying Metastatic Breast Cancer, 2016

# Harvard and MIT Approach

- **GoogLeNet AI algorithm; 27 layers and 6,000,000 parameters**
- **400 images; 270 training and 130 for testing**
- **AI was able to successfully identify cancer 92% of the time**
- **Also able to reduce human error by 85% when results were used by oncologists as a screening tool**

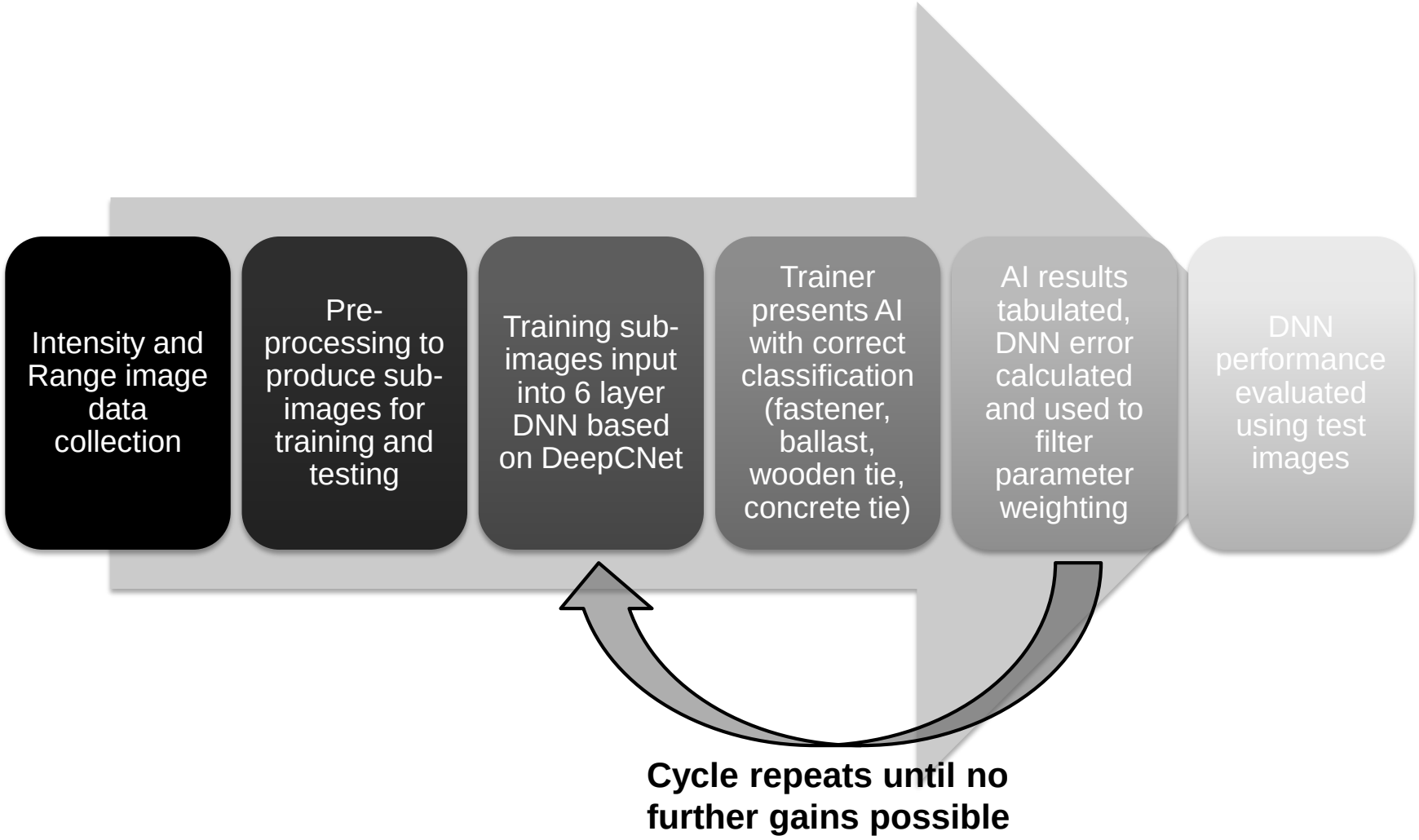



Deep Learning for Identifying Metastatic Breast Cancer, 2016

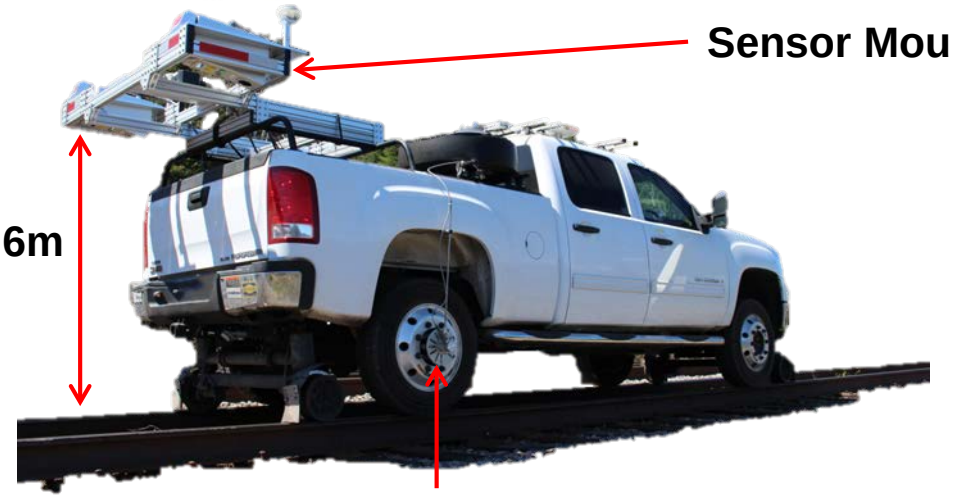
# Why Apply Deep Learning to Railroad Inspection?

- **Well-trained and experienced inspectors do an excellent job**
- **But…subjectivity, repetitiveness and acclimatization are issues**
- **Much like the field of oncology, this is an opportunity to supplement with AI**
- **Machine learning is a good start**
- **Deep learning is better however…**

# Overview of Approach



# Intensity and Range Image Data Collection

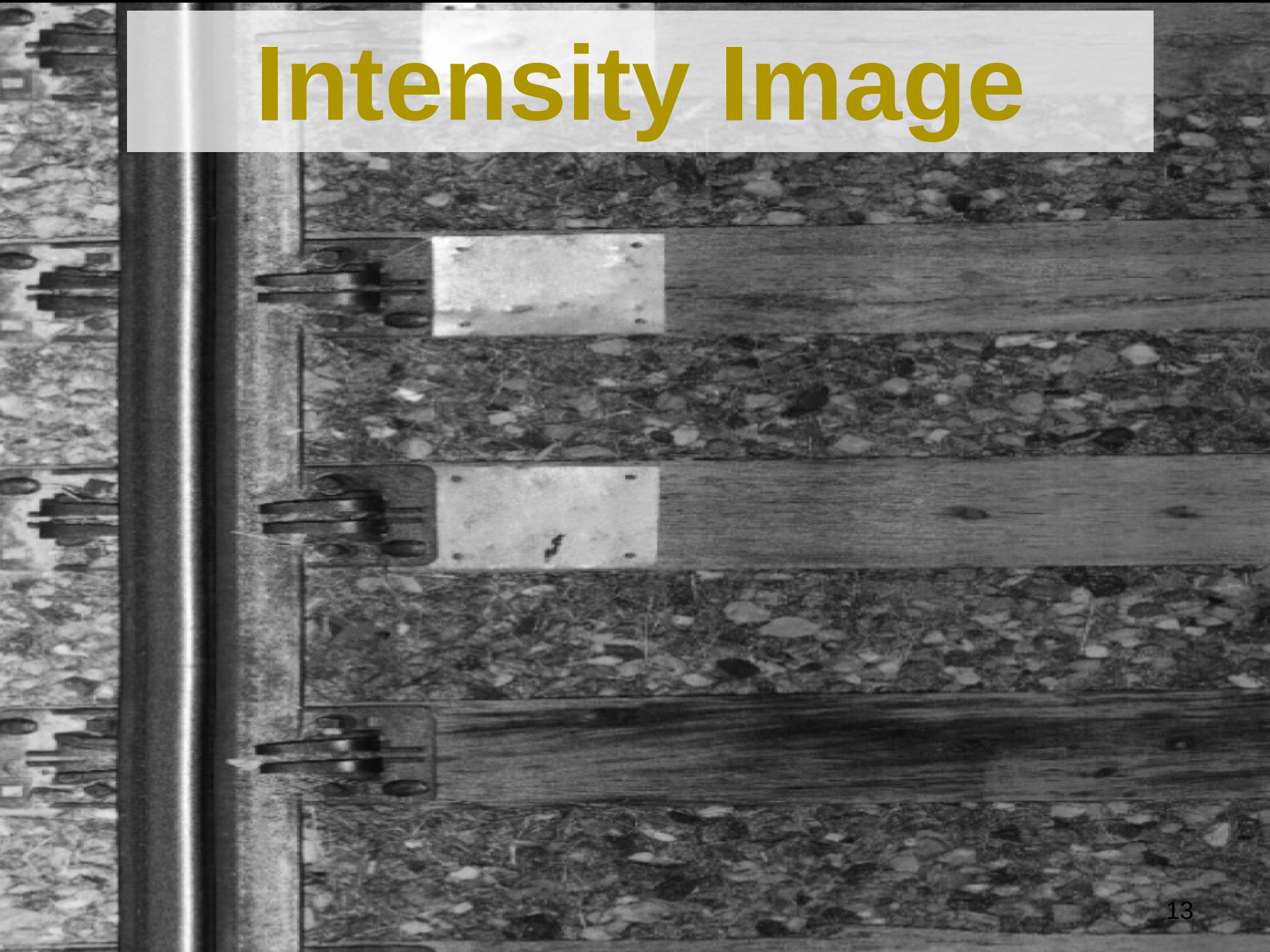



Operator's Station

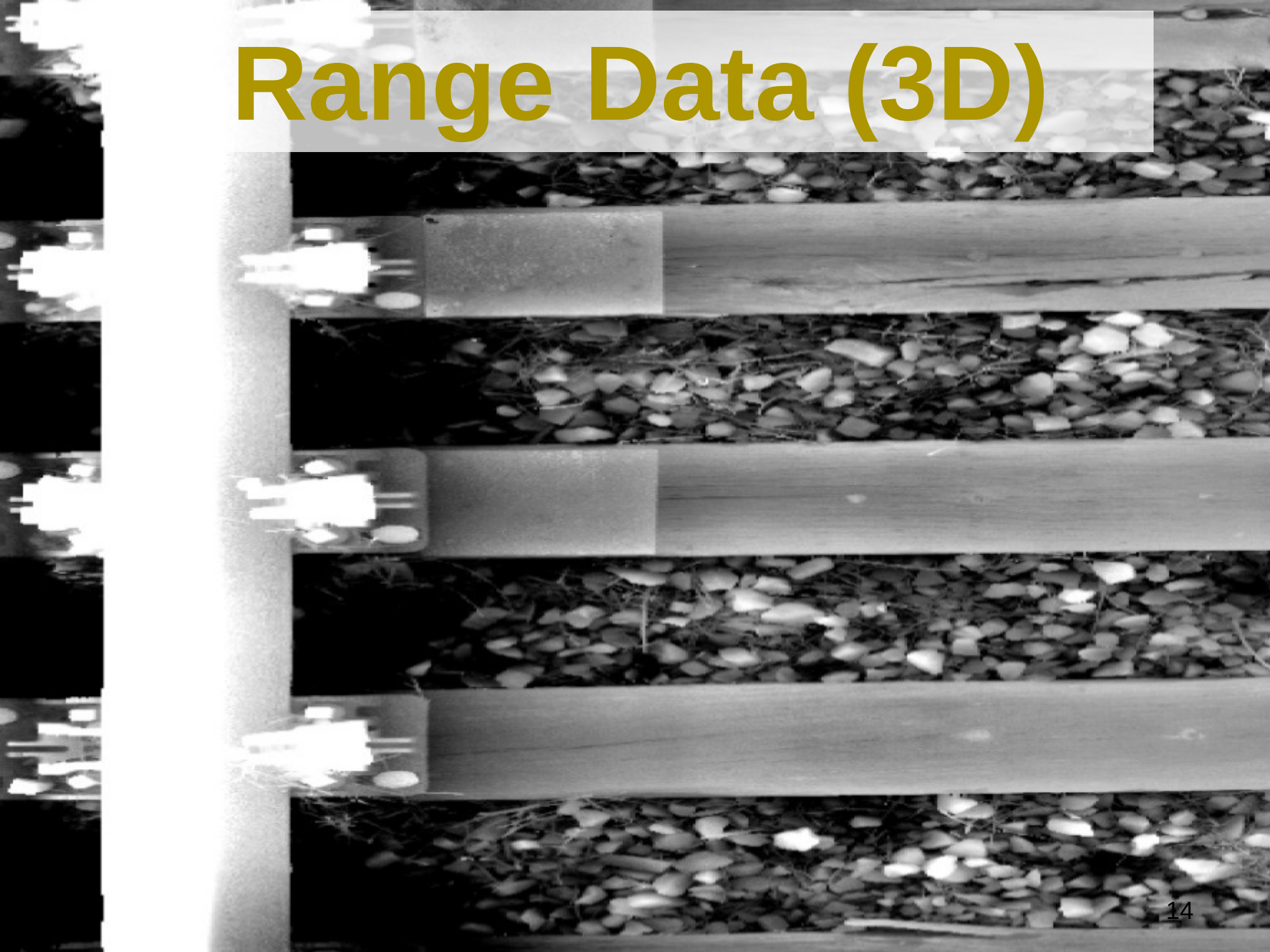
- Sensor Controller
- Data Storage PC
- Data Processing PC
- Power Invertor
- UPS

- 1 point every mm transversely and longitudinally
- 0.1mm vertical resolution
- 3.5m scan width (as tested)
- Images + 3D scan
- IMUs for motion correction

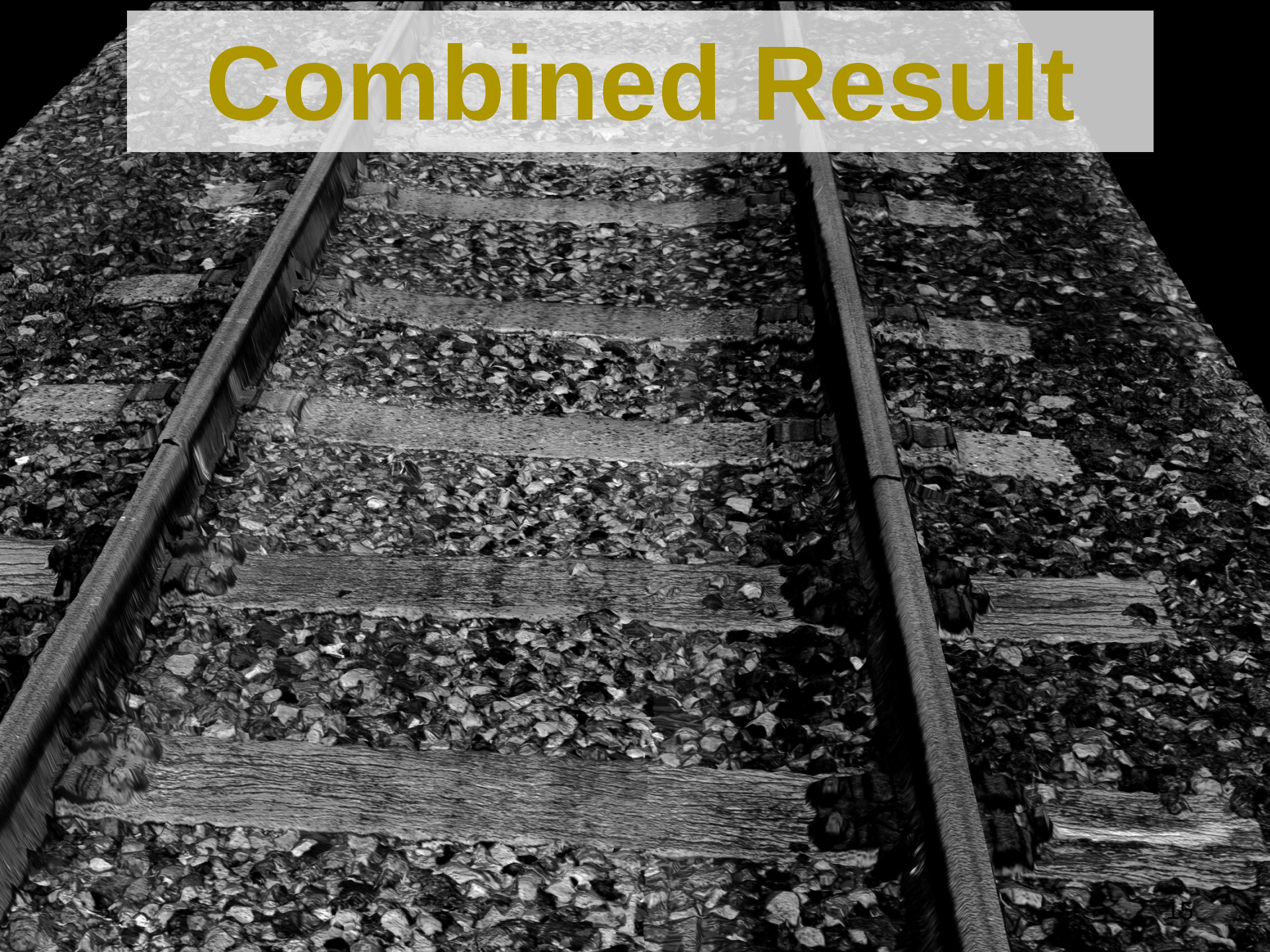
# Intensity Image



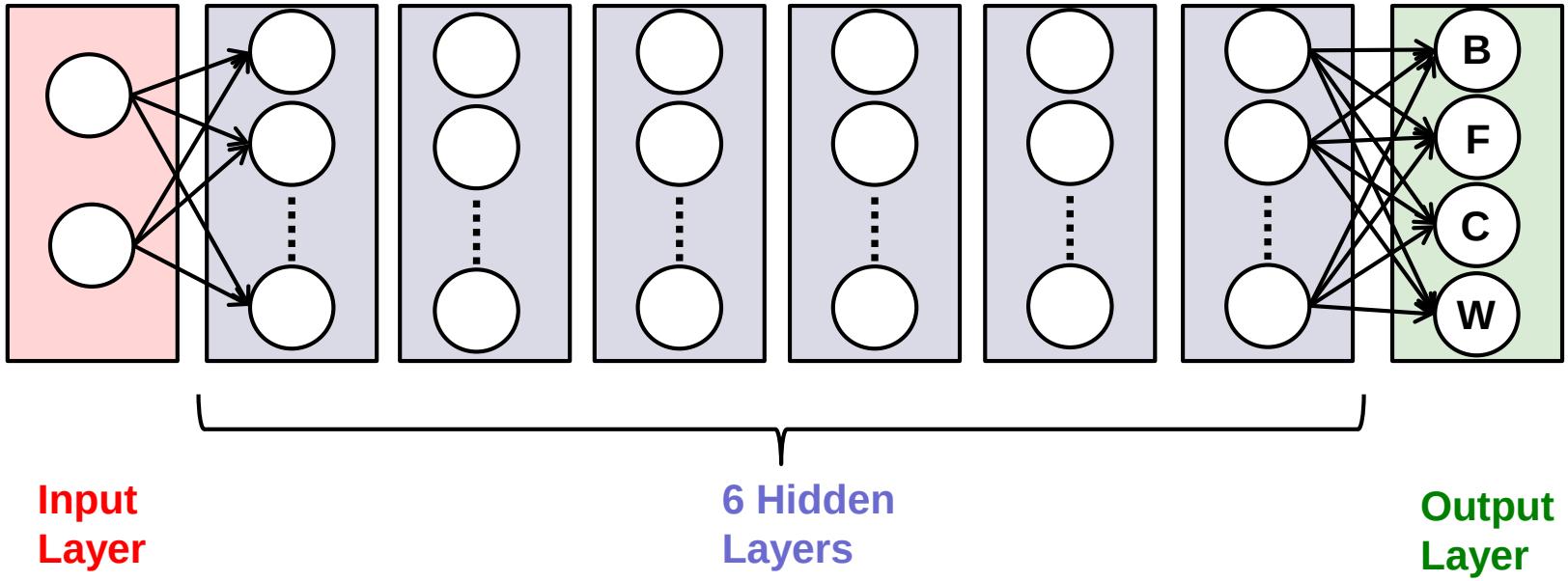
# Range Data (3D)



# Combined Result



# DNN Algorithm Details




B = Ballast F = Fastener W = Wooden Crosstie C = Concrete Crosstie

- **62,040 weights**
- **76 biases (neurons in hidden + output layers)**
- **62,116 learnable parameters (weights + biases)**

# Training the DNN

100 range and 100 intensity images; each 3.5m x 2m

Each processed to produce 2 sub images (now have 200 intensity + 200 range)

Split in half; 200 training + 200 testing

AI is presented training images + correct classification from researcher

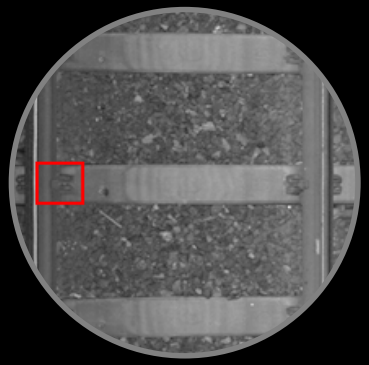
AI then presented test data (can it get the correct answer?)

107 full cycles (epochs) were performed

DNN error calculated for each cycle and used to filter parameter weighting

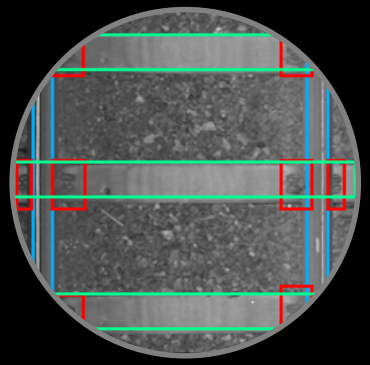
# A DNN Processing Example: Detecting a Fastener

DNN processes full images to create Sub-images



DNN processes sub-images to create raw labeled images corresponding to each asset class (fasteners, ballast, concrete tie, wooden tie)

DNN groups adjacent classified sub-images to form regions which correspond to assets



Output from image grouping is used to identify features in full resolution images

# Algorithm Performance

- **After 107 Epochs the DNN was able to correctly label/identify 100% of the images containing fasteners, ballast, and concrete ties**
- **However 1% error in labeling images containing wooden ties remained**
- **The DNN subsequently removed this error by applying a region grouping function**

| | Fastener | Ballast | Concrete Tie | Wooden Tie |
|---|---|---|---|---|
| Number of Errors | 0 | 0 | 0 | 2 |
| Percentage Error | 0% | 0% | 0% | 1% (1/99) |

# Conclusions and Commentary

- **Overall the results were promising for a small test dataset. Network-level validation is currently underway**

- **One interesting aspect of DNNs is that every network is different (nodes, layers, connections) and the actual "thought process" of the AI is constantly evolving and thus cannot be rigidly defined**

- **DNNs, like our brains, do not follow a well understood algorithm so it is not possible to patent a particular DNN nor is it possible to infringe upon existing patents with an DNN**