

Machine Learning with MATLAB at TimkenSteel

-Marc Harris



TimkenSteel Background

- ▶ Manufacturing as a whole is moving towards “Industry 4.0”
 - Data science driven predictions / decisions / automation facilitated by machine learning and optimization algorithms
- ▶ Steel industry adoption of machine learning solutions is spreading fast
- ▶ TimkenSteel adoption company-wide is focused on low capital cost fast deployment systems for:
 - Image based steel inspection facilitated by machine learning
 - Big data interpretation for process performance optimization
 - Reducing defects
 - Maximizing throughput
 - Increased automation
 - Reduced man-hours per ton
 - Capability improvements
 - New alloy development
 - Tighter process control



TimkenSteel Image Processing Projects

▶ Non-metallic inclusion image interpretation

- Human-level image interpretation
- Automation of analysis
 - Faster throughput
 - Reduction in man-hours per analysis
 - Consistency

▶ Pin Stamp Reader

- Automation of ID check process
- Reduction in man-hours per ton of steel

▶ MATLAB Toolboxes Used

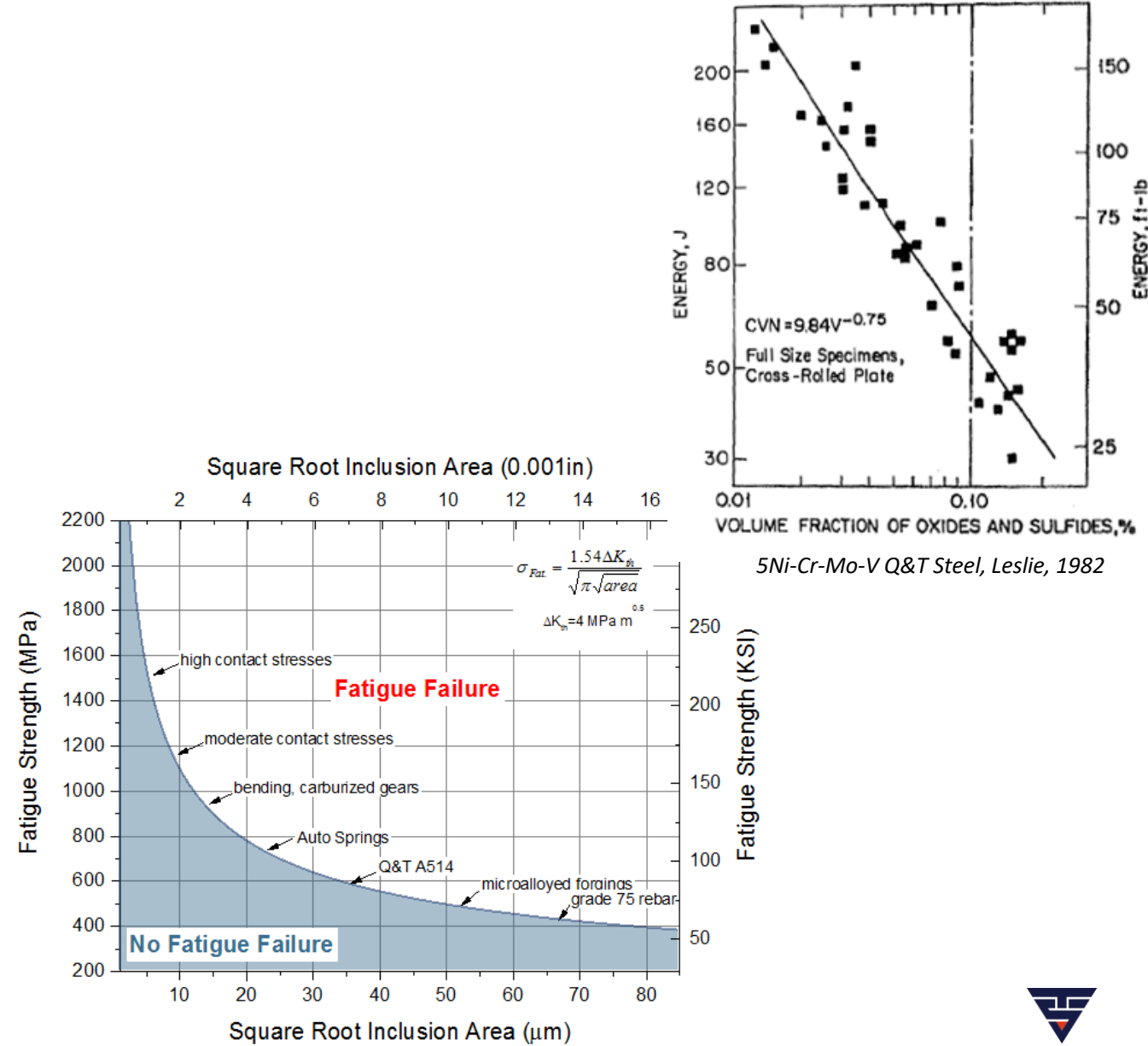
- Image Processing Toolbox
- Parallel Computing Toolbox
- Deep Learning Toolbox
- Computer Vision Toolbox



Automated Inclusion Classification

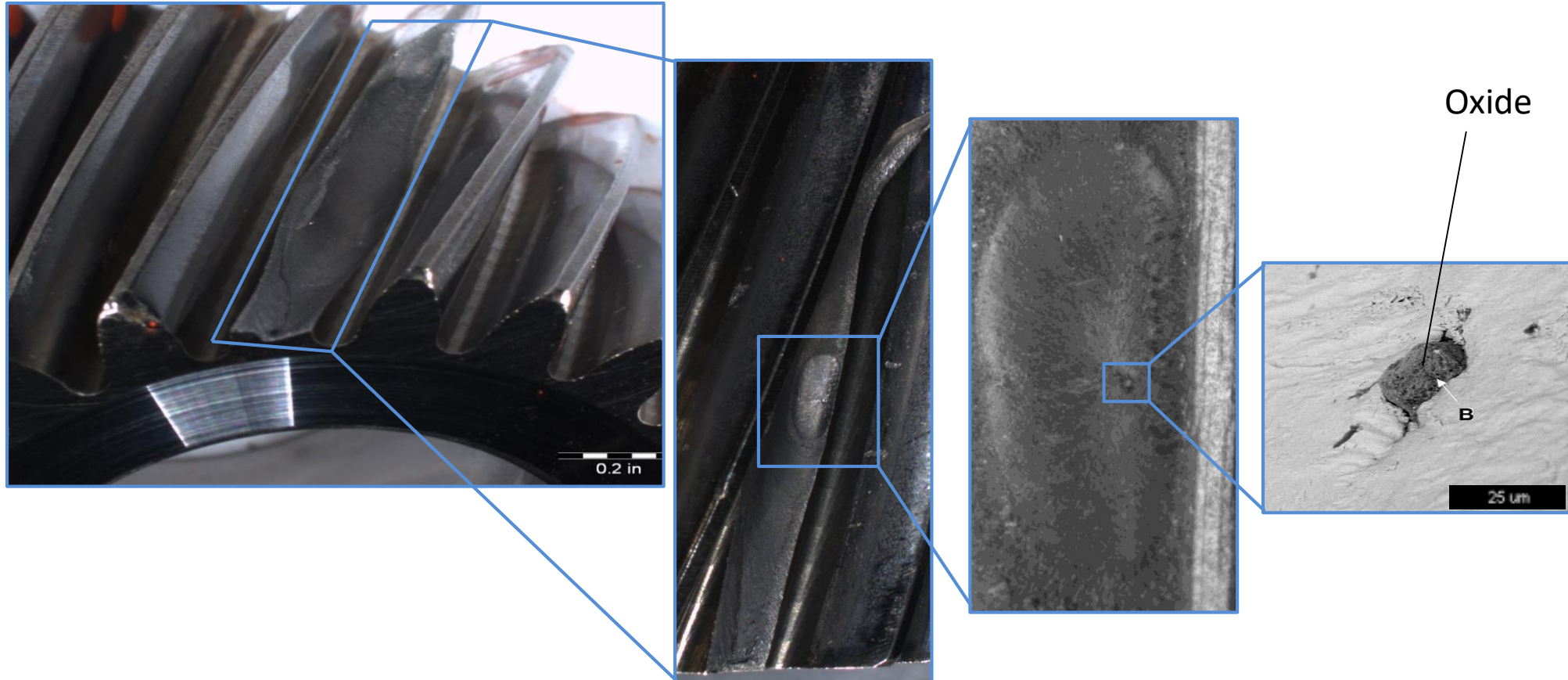
Steel Cleanness Background

- ▶ What is steel cleanness?
 - Presence of non-metallic inclusions formed thermodynamically in steel melt or through physical entrapment during melt processing
 - Size, shape, quantity, location, and chemistry are all aspects of “cleanness”
- ▶ Why do we care about steel cleanness?
 - Performance and quality are driven in large part by inclusion population
 - Cleaner steel = lighter weight for given load or more horsepower for given design
 - TimkenSteel specializes in producing extremely clean air melted steel
 - Requires focused measurement of cleanness and complete evaluation



Real World Consequences of “Dirty” Steel

- ▶ Fatigue applications such as automotive gears are highly sensitive to the inclusion content in the steel
 - Example below shows a ~20 μ m sized inclusion causing a gear tooth failure

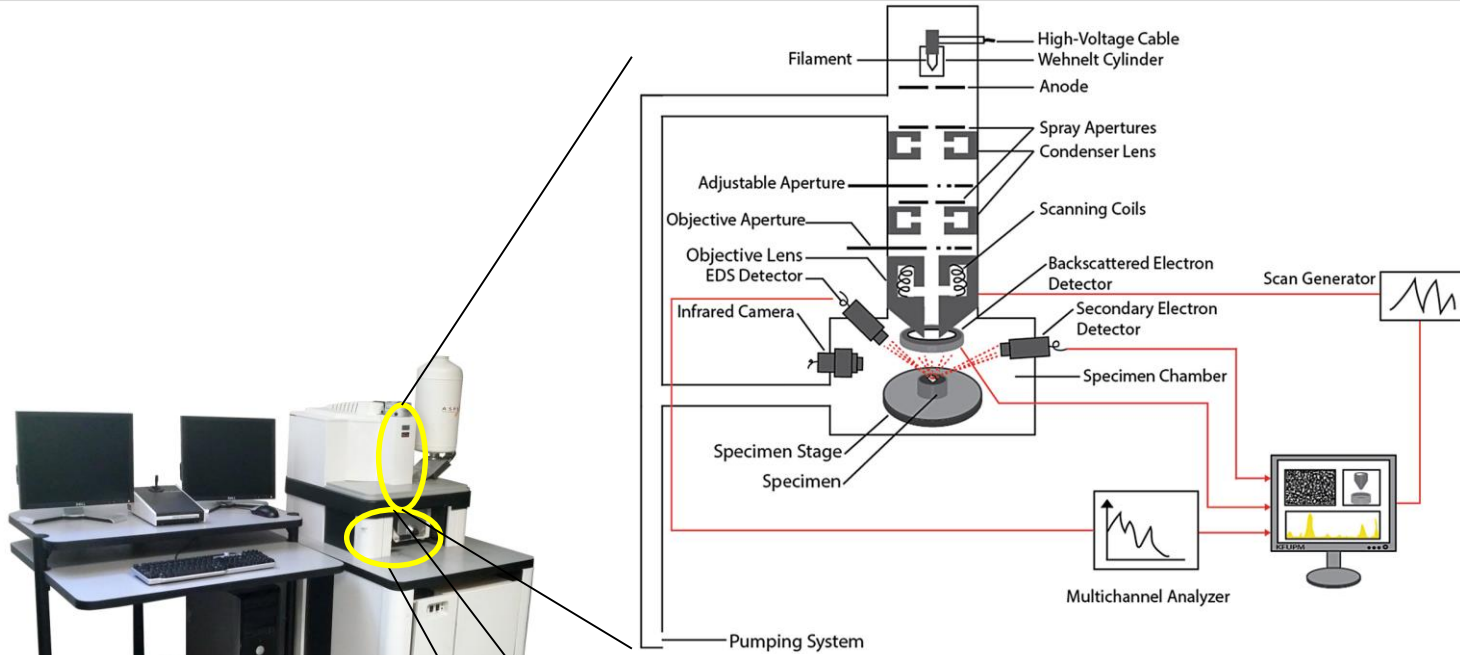


Measurement System Background

- ▶ Steel cleanliness measurement system is automated, analysis is not (yet)
 - Steel is polished to 1um finish
 - Sample is placed into an automated scanning electron microscope with energy dispersive spectroscopy
 - Sample is scanned automatically for every single inclusion >1um in size and a chemistry measured
 - Additionally coordinates, shape statistics, and other details are measured
 - Images are captured of each inclusion (1 scan = 1000's of images, one steel analysis = 32 scans)
 - Resulting dataset is evaluated using in-house software (MATLAB)
 - Engineer or trained technician has to evaluate 100's of large sized inclusion images to confirm type and remove potential contaminants that were detected from the analysis
 - 1 analysis of a steel takes 20-40 minutes depending on how many inclusions are present
 - Very manual process and can be a judgement call in some cases



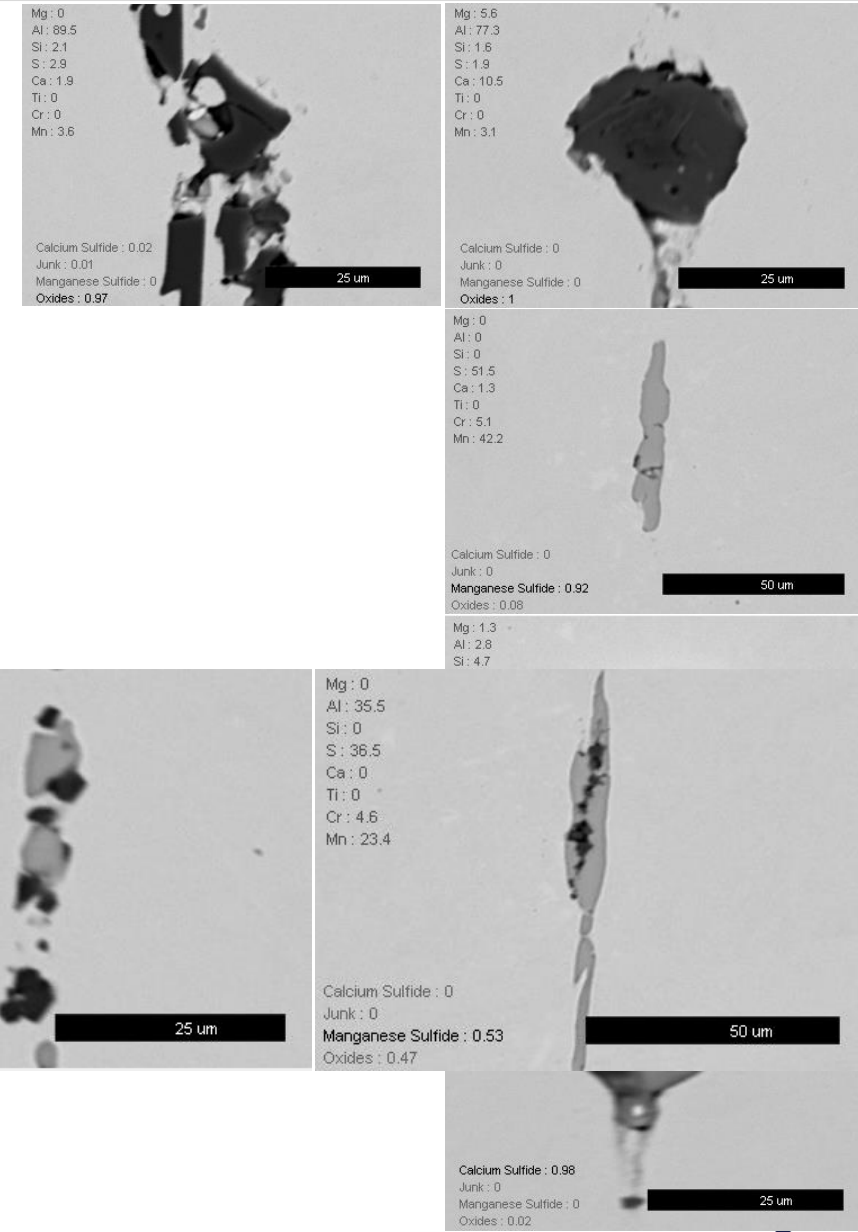
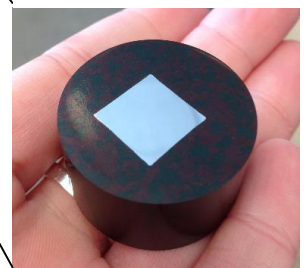
Measurement System Background



SEM - EDS

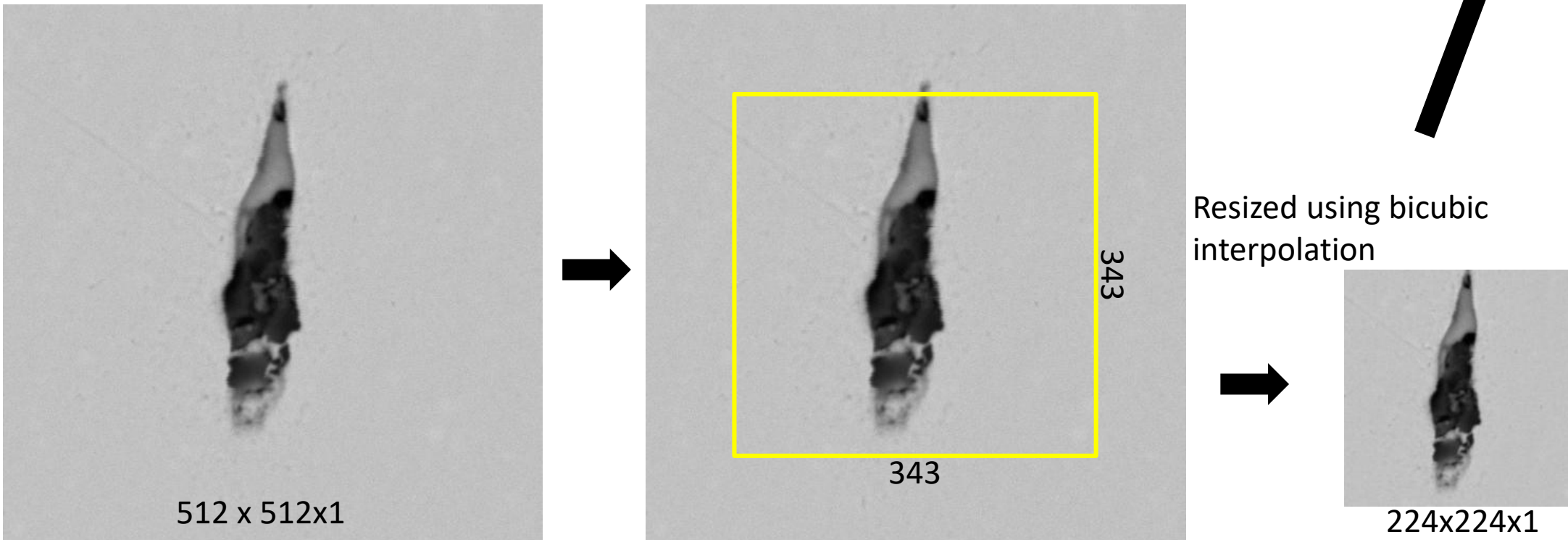
ASPEX – Automated SEM/EDS System

Polished Steel

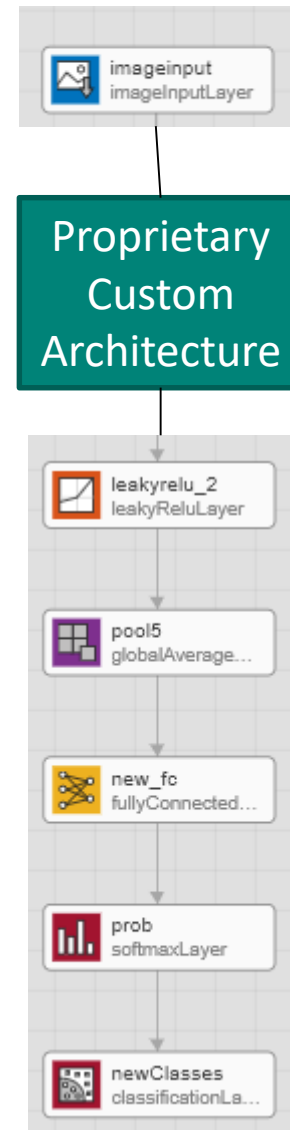


Training Process

- ▶ Raw Images resolution of 512x512 with inclusion of interest centered
 - Center crop of 2/3
 - Image resizing to 224x224
- ▶ Data augmentation
 - Random rotations from -90 to +90 degrees
 - Random brightness adjustment from -5% to +5%
 - No contrast adjustment due to calibration before each scan
- ▶ Training Method
 - Stochastic gradient descent with momentum
 - Mini-batch size of 12, 60 epochs, learning rate drop factor of 0.95 per epoch, validation set per 50 iterations



Resized using bicubic interpolation



Accuracy & Speed Testing Results

- ▶ Overall accuracy achieved of 93%
 - Human performance ~92-94%
 - Measured by taking repeated manual categorizing of same images randomly shuffled 1 week later
 - Vast majority of error in discerning CaS from Oxides and deciding final category for complex MnS-Oxide inclusions
 - Greatest error on easy to confuse CaS category
 - Accuracy for Oxides and MnS matches human specialist performance
 - Accuracy on discerning surface contaminants close to human but some further work to achieve 95%+ is in process
 - Human accuracy for discerning surface contaminants using randomly shuffled 1 week time lapse test was ~97%
- ▶ Speed of 0.012s / image achieved (~3-10 minutes per analysis)
 - Average technician/engineer takes ~2-3 seconds/image
 - System fully automated
 - Completely consistent (human fluctuation and variance does not impact or confound cleanness results)

Training Image Count Breakdown

Junk: 261
 Oxides: 350
 MnS: 364
 CaS: 212

Training + Test Set Confusion Matrix

Pred. \ Actual	Junk	Oxides	Manganese Sulfide	Calcium Sulfide
Junk	273	8	6	7
Oxides	22	1700	87	53
Manganese Sulfide	7	81	2405	37
Calcium Sulfide	2	25	9	203

If Model says X, what is the probability that is true?

Model Prediction	Accuracy
Junk	89.8%
Oxides	93.7%
MnS	95.9%
CaS	67.7%
Anything	93.0%



Final Result for User

▶ Practically invisible to user – auto classification

- Analysis is practically a button press
- Human engineer/specialist accuracy

The screenshot displays six particle analysis results arranged in a 2x3 grid. Each result consists of a grayscale image of a particle, a list of elemental concentrations, classification labels, and a scale bar. Below each image is a dropdown menu showing the selected classification.

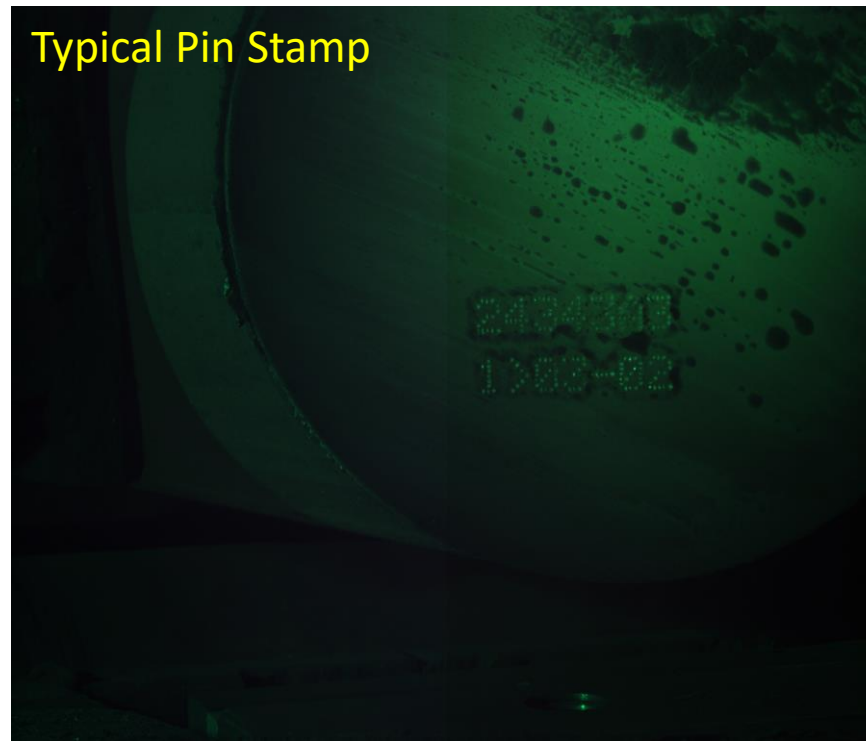
Row	Column	Elemental Concentrations	Classification Labels	Scale Bar	Selected Classification
1	1	Mg: 0 Al: 0 Si: 0 S: 46.9 Ca: 51.8 Ti: 0 Cr: 1.3 Mn: 0	Calcium Sulfide: 0 Junk: 1 Manganese Sulfide: 0 Oxides: 0	25 um	Junk
1	2	Mg: 0 Al: 2.8 Si: 0 S: 36.5 Ca: 0 Ti: 0 Cr: 2.2 Mn: 58.4	Calcium Sulfide: 0 Junk: 0 Manganese Sulfide: 1 Oxides: 0	25 um	Manganese Sulfide
1	3	Mg: 0 Al: 0 Si: 0 S: 36.5 Ca: 0 Ti: 0 Cr: 1 Mn: 62.5	Calcium Sulfide: 0 Junk: 0 Manganese Sulfide: 1 Oxides: 0	25 um	Manganese Sulfide
2	1	Mg: 0 Al: 2 Si: 0 S: 35.4 Ca: 0 Ti: 0 Cr: 0 Mn: 62.6	Calcium Sulfide: 0 Junk: 0 Manganese Sulfide: 1 Oxides: 0	50 um	Manganese Sulfide
2	2	Mg: 0 Al: 4 Si: 0 S: 34.5 Ca: 0 Ti: 0 Cr: 1.4 Mn: 60.1	Calcium Sulfide: 0 Junk: 0 Manganese Sulfide: 1 Oxides: 0	25 um	Manganese Sulfide
2	3	Mg: 7.5 Al: 56.1 Si: 1.2 S: 10.1 Ca: 19.3 Ti: 0 Cr: 0 Mn: 5.9	Calcium Sulfide: 0 Junk: 0 Manganese Sulfide: 0 Oxides: 1	10 um	Oxides



Pin Stamp Reader (in process)

Background

- ▶ All steel must be tracked + double checked to ensure no product mixes occur or can occur
 - Product mix is when a “mix up” happens causing a different steel grade/size/order to be processed incorrectly as a result
 - Rare but very costly mistake
- ▶ Current method is human double checking a pin-stamp + piece tracking software system
- ▶ Automating this process would alleviate tens of thousands of dollars in man-hours



Initial Failure with Simple Crop-Scan-Classification Approach

- ▶ Initial attempt was using Resnet101 and sliding crop window
 - Crop -> Forward pass -> Crop Scan -> Forward Pass process took 15 seconds for ~50% accuracy of centering pin stamp
 - Further analysis using pre-spaced crop windows (see right) resulted in overall accuracy <20% due to crop misalignment
 - Balancing drastic speed reduction versus accuracy – must maintain <10 second pace per image
 - Reducing step size improved accuracy but drove computation time to minutes
- ▶ Determined best solution is YOLOv2 version of convolutional neural net for pre-crop system

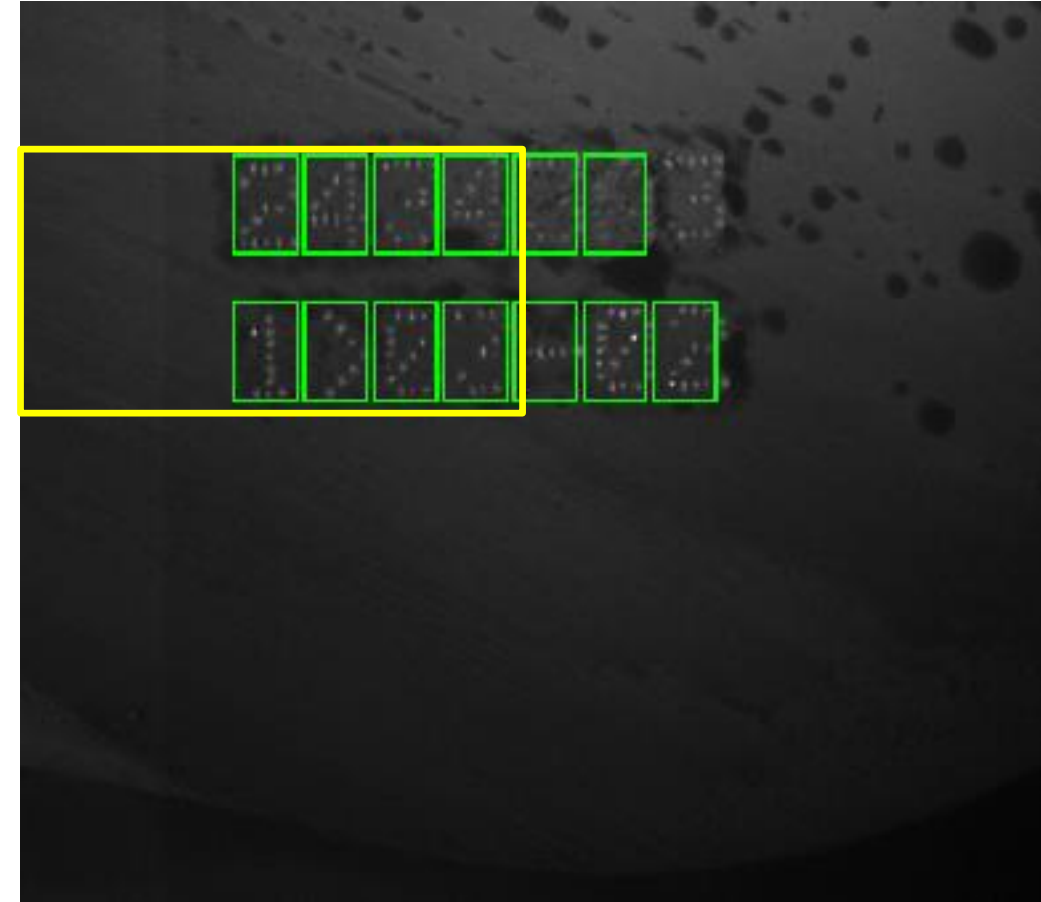
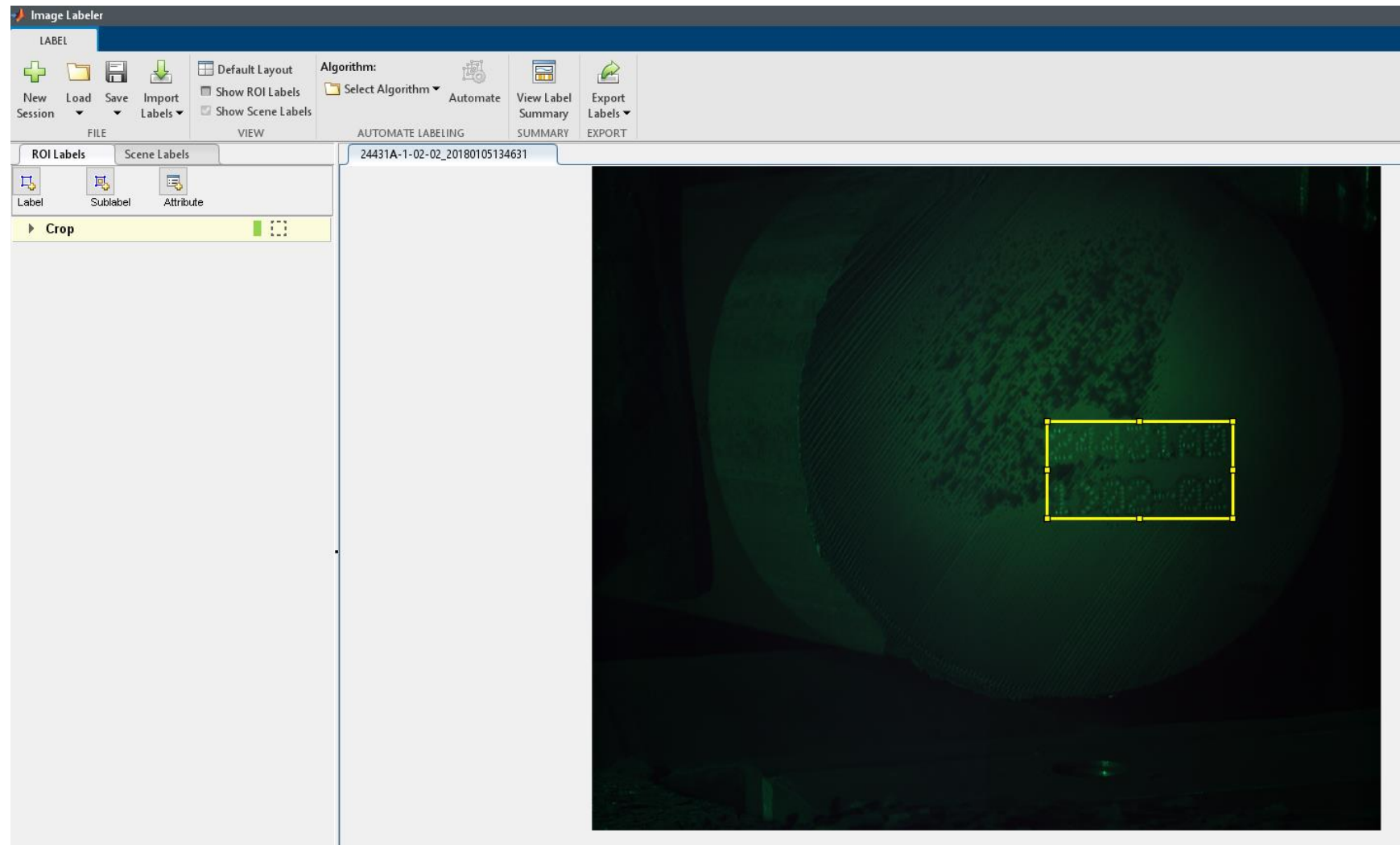


Image Labeling – YOLOv2 CNN

- ▶ Training dataset created by manually drawing bounding box of each object of interest
 - In the case of initial cropping just a centered section of the pin-stamp



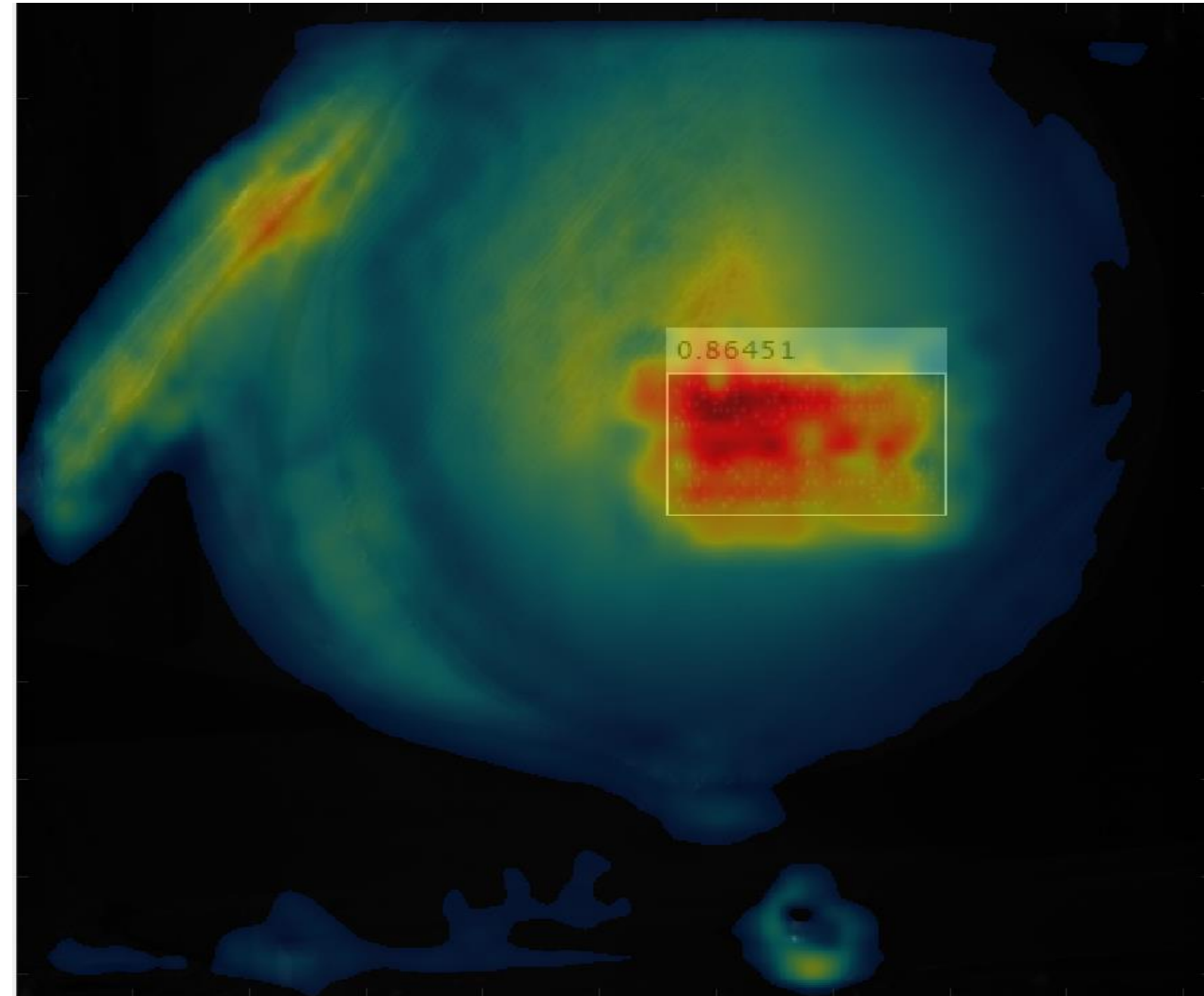
Training Process

- ▶ Training set was 168 images
 - Raw image resolution 2432x2050
 - Resized to 512x512
- ▶ CNN architecture was 25 layer YOLOv2
- ▶ Training was performed using stochastic gradient descent with piecewise learning rate reduction
- ▶ Data augmentation was used to reduce overfitting
 - Brightness +/- 40%
 - Contrast +/- 40%
 - Saturation +/- 10%
 - No reflections, +/- 10% scale, +/- 100 pixel translation in any direction, +/- 2 degree rotation



Results on Test Set

- ▶ 145/286 test images (never seen before) were *perfectly* cropped
 - 51% accuracy in 0.1-0.2 second run time
- ▶ 141/286 test images incorrectly cropped
 - 138 properly sized and existing on image
 - 70 images off by only 1-2 digits in horizontal direction
 - 30-40 images extremely noisy or vastly different from training dataset
 - No Expectation for success (not even for human)
 - To be retrained mixing in some more noisy images
 - 33 images were truly errored (11.5% error)
- ▶ Available dataset is >10,000 images
 - Future work is expanding training dataset for more variance



Future Work – Pin Stamper

- ▶ YOLOv2 system had <1 week turnaround from learning to use Matlab module to a working model & surpassed previous record performance with only 168 training images
 - Retrain using larger dataset
 - Include more variance in training set to account for noise
 - Experiment with architecture for improved accuracy/speed ratio



Deployment

TIMKEN STEEL 

Two Methods Used

▶ MATLAB Compile to .NET dll

- Advantages
 - Fully contained pre-processing and model interpretation
 - Ease of deployment
- Disadvantage
 - Memory usage / speed
 - MATLAB run time .dll requirements
 - Cross platform compatibility

▶ Export to ONNX (consumed in C#)

- Advantages
 - Widely used system that is recognized in large variety of languages from c# to python
 - Model upkeep/translation in practically any common language is easy
 - Generalized deployment minimizes any code refactoring
- Disadvantage
 - More difficult initial deployment code
 - Only a one time hit since generalized .ONNX model interpretation automatically handles variety of models types/shapes/sizes



Conclusions

- ▶ TimkenSteel has seen several benefits from adopting a machine learning approach to numerous problems
 - Rapid alloy development surpassing known mechanical properties
 - Automation of previously manual processes saving cost on man-hours
 - Faster throughput in manufacturing processes through optimization
 - “Free” steel inspection using only already-existing cameras with deep learning
- ▶ MATLAB provides a rapid-prototyping and development platform that makes deep learning/image processing simple
 - Speed of idea to finished model is magnitudes faster than the organization of the data itself
 - First deep learning system took 4 days to learn, produce, and have an accurate model
 - Subsequent systems have been more complex and expanding our capabilities with very little time needed for actually creating/training the CNN
 - Other systems not as user-friendly, slower to develop, and non-visual
 - Have looked at python, ML.net, and TensorFlow



Thank you!

Questions?

