Deep Learning for Railroad Inspection – Phase 2

Mario Talbot, John Laurent, Richard Fox-Ivey

Pavemetrics Systems Inc.

Table of Contents

Abstract

Railway networks around the world are an important part of the transportation network and represent billions of dollars of investment. Poorly maintained networks negatively impact asset longevity, schedule performance and pose a serious threat to safety. In order to safeguard against these risks, Railroads typically inspect 100% of their mainline network at least annually and key locations even more frequently. Railroad inspection has traditionally been a manual process with inspectors walking the track or driving slowly in a high-rail vehicle to visually spot problems. This practice is very costly, time consuming, impacts schedule performance (due to the need for track possession), and puts staff at risk. While there have been some recent attempts to modernize the inspection process through the adoption of machine-vision technologies, these technologies are often still reliant on human inspectors manually reviewing images in order to spot defects. Manual review of images suffers from many of the same problems as manual inspections do: it is time consuming, subjective as opposed to being objective, and requires significant amounts of labor. This paper builds on prior work to develop a Deep Neural Network that can automatically identify key railway components as a step in the process of automating rail inspection in an effort to overcome the limitations of traditional methods. This new study adds the identification of new railway components (Tie Plates) as well as the automated assessment of their condition.

Keywords: Artificial Intelligence, Neural Network, Machine Learning, Deep Learning, Deep Neural Networks, Railway, Safety, Inspection, 3D Laser Triangulation, Fasteners, Ties, Ballast, Tie Plate Inspection, Automated Assessment

1 Introduction

Railway networks around the world are an important part of the transportation network and represent billions of dollars of investment. Poorly maintained networks negatively impact asset longevity, schedule performance and pose a serious threat to safety. In order to safeguard against these risks, Railroads typically inspect 100% of their mainline network at least annually and key locations even more frequently. Railroad inspection has traditionally been a manual process with inspectors walking the track or driving slowly in a high-rail vehicle to visually spot problems.

This practice is very costly, time consuming, impacts schedule performance (due to the need for track possession), and puts staff at risk. While there have been some recent attempts to modernize the inspection process through the adoption of machine-vision technologies, these technologies are often still reliant on human inspectors manually reviewing images in order to spot defects. Manual review of images suffers from many of the same problems as manual inspections do: it is time consuming, subjective as opposed to being objective, and requires significant amounts of labor.

This paper will explore a new approach that has the potential to overcome these limitations using Deep Learning algorithms, specifically a Deep Neural Network (DNN), to automatically inspect 3D Laser Triangulation images.

3D laser triangulation captures a both a high-resolution image (2D) and a 3D point cloud of the entire track area and can be used at revenue speeds, day or night.

DNN is a type of machine learning wherein the computer develops a solution to a complex problem in a way that is similar to how humans learn (using a neural network). Deep

Learning is well-suited to image analysis and has even been demonstrated to improve the accuracy of cancer detection by oncologists when used to analyze images of lymph nodes.

This paper includes an overview of the principle of operation of 3D laser triangulation sensors as well as the field of Deep Learning. Sensor scanning frequency, resolution and accuracy are discussed along with data storage requirements and formats. The Deep Learning algorithm training process is also discussed and examples of Deep Learning image classification from both outside the rail industry, as well as inside the industry, are presented.

2 Artificial Intelligence, Machine Learning and Deep Learning

2.1 Artificial Intelligence Background

John McCarthy (1927-2011), an American computer and cognitive scientist, coined the term Artificial Intelligence (AI) at the Dartmouth Conference in 1956 (John McCarthy (computer scientist), 2018). Since that time significant advances have been made in the field of AI, particularly in the area of Machine Learning (ML), and more recently in the sub discipline of Deep Learning (DL).

Broadly speaking, ML relies on the use of human-developed algorithms to parse data, learn from the data, and then apply that knowledge to make subsequent decisions. One application of ML is the recommendation of new songs to streaming music subscribers based on matches with their historical listening preferences (Ciocca, 2017). However, while ML can be used to crunch large volumes of data in order to make useful recommendations, it relies upon a human designer to spell-out the logic behind those recommendations and, as such, it is ultimately limited to solving problems which the designer already knows how to solve.

DL, a subset of ML, has the potential to outstrip ML due to its ability to develop its own methods of analysis, much in the same way that a human brain does, through the use of Artificial

Neural Networks (ANN). DL algorithms also have the advantage of not requiring re-training as new examples or classes are added.

In ANN individual elements in the network are referred to as neurons and they are arranged in layers (input, hidden layers and the output layer) with connections between neurons being referred to as synapses (Figure 1 - Neural Network, (Stanford University, 2017)).

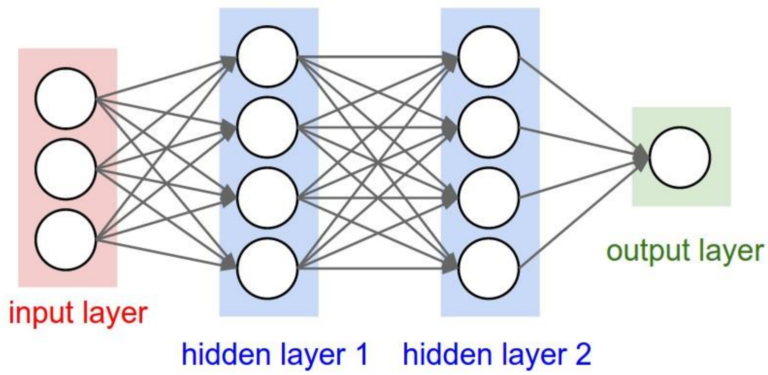

Figure 1 - Neural Network

The size and complexity of a neural network is typically defined in terms of its number of hidden layers and its total number of learnable parameters (Stanford University, 2017). Deep networks are networks which contain multiple hidden layers as opposed to just one or two.

The total number of parameters in an ANN is determined by summing its weights and biases. Weights are determined by the number of possible permutations of its input, hidden layers and output layer. Biases are determined by summing the neurons in its hidden layers and the output layer. As an example, the ANN in Figure 1 contains a total of 9 neurons (not counting the inputs), 32 weights, 9 biases and 41 learnable parameters.

In 2018 the science behind DL already impacts millions of people on a daily basis as it is behind both the recommendations that Netflix makes regarding what to watch next (Alex Chen, 2014), and the facial recognition Facebook performs to tag friends in uploaded images (Marr, 4 Mind-Blowing Ways Facebook Uses Artificial Intelligence, 2016).

The ability of DL to solve problems which the designer does not yet know how to solve is one of its key advantages. An example of this advantage can be found in Google's AlphaGo computer program. The DL algorithms in AlphaGo have enabled it to learn how to play the 2,500+ year old Chinese strategy game *Go* by playing against human opponents. Over time the AlphaGo DL algorithm has been able to improve its playing ability (essentially learn), through repeated matches, to the point where it has been able to beat a number of the world's top players (The story of AlphaGo so far, 2018). This feat would not have been possible using ML only as a ML computer would be limited to consulting a pre-defined list of moves and recommended counter-moves rather than developing its own strategy for matches.

Directly relevant to the topic at hand, is the application of DL in the medical field to solve the problem of cancer cell identification in lymph node images. In 2016 a DL team from the Harvard Medical School's Beth Israel Deaconess Medical Center (BIDMC) and the Michigan Institute of Technology (MIT) was able to reduce the human error rate in cancer diagnosis by an incredible 85 percent and successfully identify cancer 92 percent of the time (Kontzer, 2016).

In theory there is no limit to the application of DL due to its ability to tackle novel problems by developing its own solutions. It is with this in mind that the research team turned to the challenging problem of developing DL strategies for the automated inspection of railway asset condition and safety.

2.2 A Deep Learning Case Study: Identifying Cancer Cells in Lymph Nodes

The field of Pathology is responsible for providing accurate and repeatable diagnoses of disease in order to help patients make informed decisions about treatment and management. However the task of disease identification primarily relies upon the qualitative assessment of cells under a microscope. Even for well-trained pathologists this is a time consuming and strenuous task due to lack of standardization, diagnostic errors, and the significant cognitive load associated with manually evaluating images (Dayong Wang, 2016).

In 2016 a team from Harvard's Beth Israel Deaconess Medical Center and the Massachusetts Institute of Technology (The Harvard and MIT Team) was successful in placing first for both competition categories at the prestigious Camelyon Grand Challenge. Their project involved the development of a DL algorithm to assist pathologists in the identification of metastatic breast cancer cells.

Their approach involved "millions of training patches to train a deep convolutional neural network to make patch-level predictions to discriminate tumor-patches from normal-patches" (Dayong Wang, 2016).

The algorithm development process involved the use of a dataset of 400 images containing both cancerous and non-cancerous cells (identified by a pathologist). Of the 400 images 270 were used for algorithm training purposes and the remaining 130 were used to test the resulting algorithms. During the training process slide images were combined with ground truth data in order to create both positive and negative, 256 x 256 pixel, sample images which were used to teach the DL algorithm how to distinguish between cancerous and non-cancerous cells. Following the training process the DL algorithm was tested against the 130 test images in order to determine algorithm performance (Figure 2 - Framework of Cancer Metastases Detection).

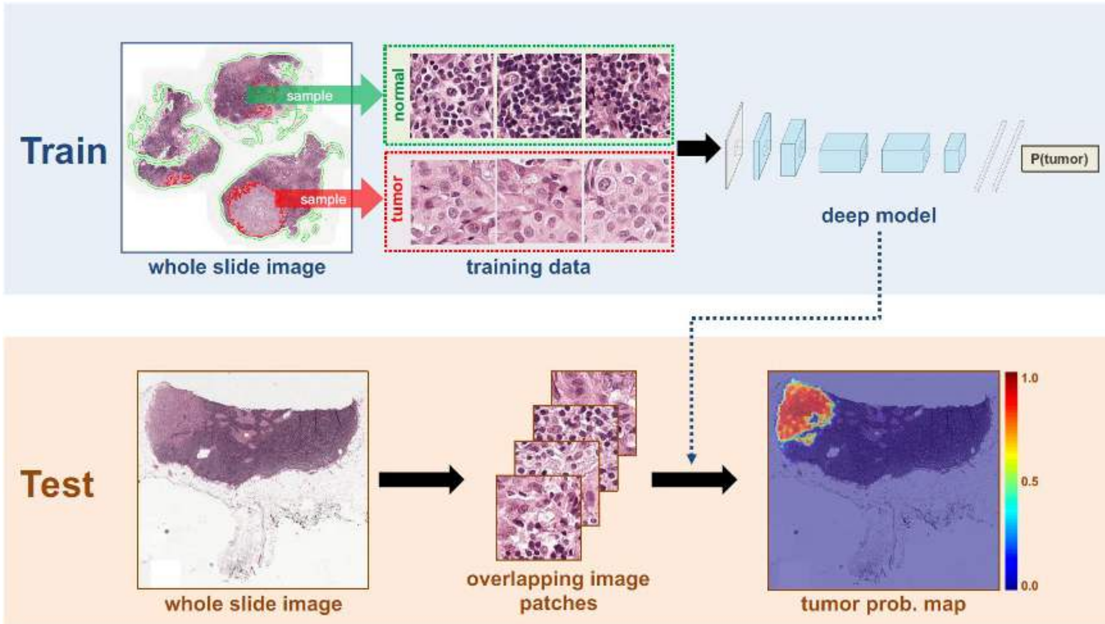

Figure 2 - Framework of Cancer Metastases Detection

The performance of four popular deep learning architectures, including GoogLeNet, AlexNet, VGG16 and FaceNet, were evaluated by the team. Of the four architectures tested, the two deeper architectures, GoogLeNet and VGG16, produced the best results (Dayong Wang, 2016).

The team ultimately settled on the use of GoogLeNet, which offers 27 layers and more than 6 million parameters, for its fast performance and stability. While the performance of the DL algorithm at the time of the competition was not yet able to best a human pathologist (it yielded an error rate of 7.5% versus 3.4%) the algorithm *was* able to reduce the pathologist's error rate to just 0.52% when its predictions were combined with the pathologist's (Dayong Wang, 2016).

It is with this success in mind that the research team decided to adopt a similar approach in the development of DL algorithms for the analysis of railway inspection images and the identification of railway maintenance and safety defects.

3 Applying Deep Learning to Railway Inspection

3.1 Inspection Challenges in the Railway Industry

Railway networks around the world represent billions of dollars of investment. Poorly maintained networks negatively impact asset longevity, schedule performance and pose a serious threat to safety. In order to safeguard against these risks, Railroads typically inspect on hundred (100) percent of their mainline network at least annually, and key locations even more frequently.

However, aside from the use of rail geometry measurement cars, the average railroad inspection is largely a manual process with inspectors walking the track or driving slowly in a high-rail vehicle to visually spot problems. This practice is very costly, time consuming, impacts schedule performance (due to the need for possession), and puts staff at risk.

In contrast, modern image-based approaches to railway inspection seek to create a permanent visual record of railway conditions which can be subsequently analyzed by trained inspectors from the comfort and safety of an office environment. Inspectors are trained to spot potential problems in the railway network visually, including: rail surface damage, damaged crossties (sleepers) and missing fasteners.

However, the task is not an easy one as rail networks are typically not homogenous in nature and can contain a wide variety of rail types and weights, fastener types and configurations, crosstie materials and conditions, special track work, and other elements which the inspector must identify and evaluate.

While there are indeed a great many advantages to the use of human expertise for visual inspection, there are also a number of challenges. Key among them is the subjective nature of human inspection in terms of data quality with no two inspectors “seeing” the same railway condition in exactly the same way. Compounding this problem is the repetitive nature of the task which tends to lower inspector performance as their working shift progresses. Lastly there is the impact of both personal and professional pressures on the quality of results from factors such as work deadlines, personal health, etc.

In recent years the industry has responded to this challenge by attempting to develop computer-based image processing techniques to supplement or replace the efforts of human inspectors. Invariably these approaches involve ML techniques which rely upon the designer defining the elements which the computer algorithm should identify. While some successes have been realized using this approach, the complex nature of the railway environment makes it very difficult to program for all instances.

DL has the potential to address the short comings of both human-based inspection and ML-based inspection by leveraging the best qualities of both approaches. Much in the same way that DL has shown potential to assist pathologists with the challenging task of image analysis in the medical field, the authors believe that inspectors in the railway industry can be likewise benefit from the power of DL.

3.2 Prior Research Performed by the Project Team

The task of automatic identification of railway assets was selected first in order to evaluate the potential for DNN to assist in railway inspection. This was deemed a logical first step as later research into the reporting of missing or damaged components, for example, would require existing features to first be detected and identified.

The first step in the process was to capture intensity and range images using a 3D Laser Triangulation sensor, and to correct them for the effects of vehicle motion. Captured images were then fed into the DNN and 35 x 35 pixel sub images were created to serve as inputs for algorithm training and testing. Sub images were then processed in order to create raw labeled images corresponding to four railway asset types; fasteners, ballast, wooden ties and concrete ties. Adjacent classified sub images with the same classification where then grouped together in order form regions corresponding to assets. Output from the image grouping process was then used to identify features present in the full resolution images (Figure 3 - High-Level Approach):

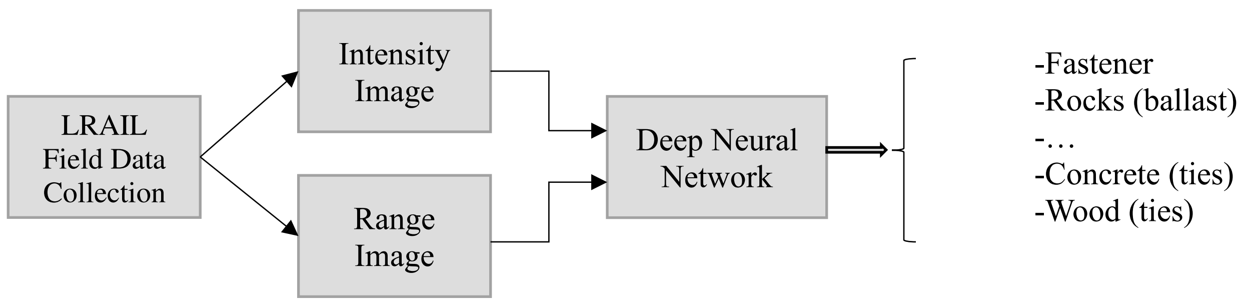

Figure 3 - High-Level Approach

3.3 Present Research Focus

Prior work focused on demonstrating the potential for a DNN to assist in automated railway inspection by properly identifying railway components including fasteners, ballast, wooden ties and concrete ties.

The focus of the present project was to expand upon prior work by training the DNN to identify a new component, Tie Plates, as well as performing an automated assessment the installation of the Tie Plate in order to flag faulty installations. Tie plates which where miss-aligned (twisted), shifted (to the left or to the right) or no longer in contact with the foot of the rail were flagged as faulty.

3.4 Image Capture Equipment

Pavemetrics' Laser Rail Inspection System (LRAIL) was used for image capture. The LRAIL is a 3D Laser Triangulation sensor which combines pulsed high-power invisible laser line projectors, custom filters, and synchronized cameras to capture a high-resolution intensity image and 3D range profile of the railway trackbed. Laser light is used to illuminate railway surfaces and high-speed cameras are used to capture images of the projected light including its intensity and 3D shape (Figure 4 - Laser Triangulation Principle of Operation).

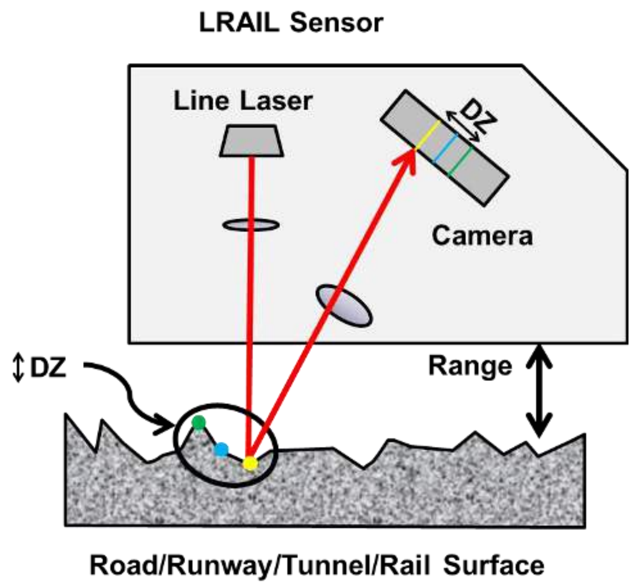

Figure 4 - Laser Triangulation Principle of Operation

For this project a specialized high-rail trailer was constructed and fitted with the LRAIL sensor technology, battery-bank power supply, data storage computers, GPS receivers and related hardware (Figure 5 - LRAIL Self-contained Test Trailer and Figure 6 - Onboard Computing and Power Supply).

Figure 5 - LRAIL Self-contained Test Trailer

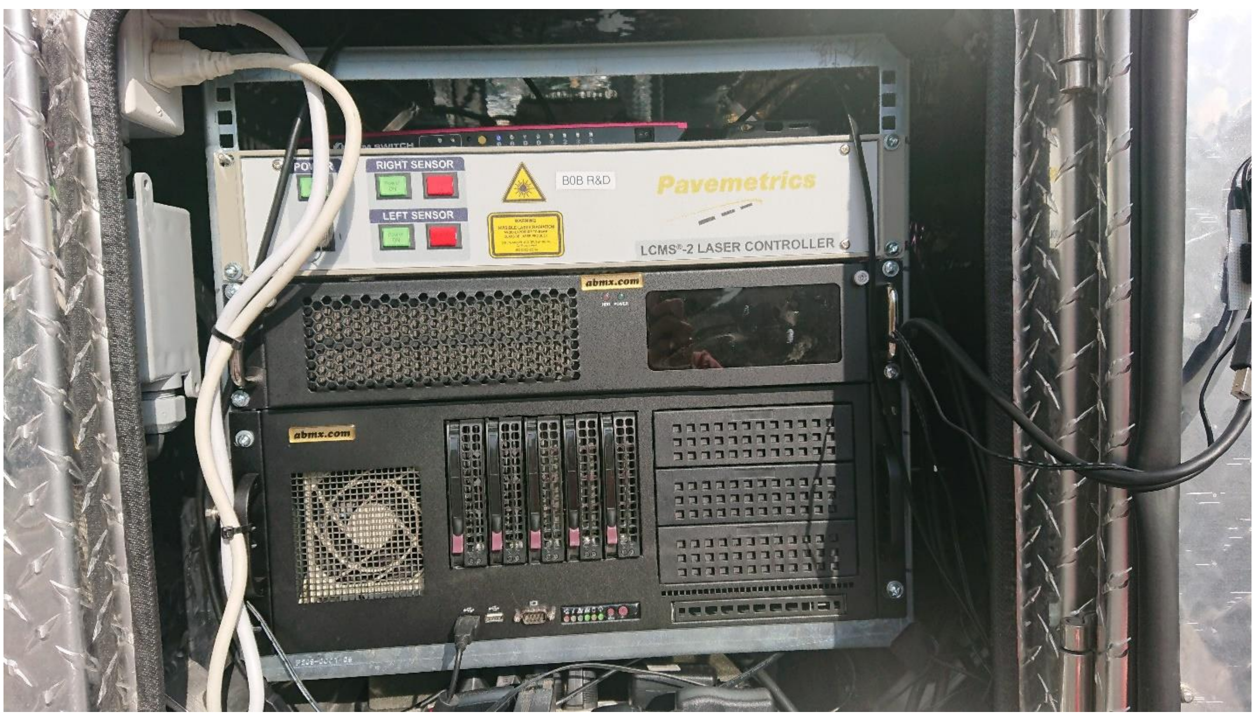

Figure 6 - Onboard Computing and Power Supply

This self-contained and self-powered configuration allowed the LRAIL to be quickly deployed to track for testing without the need for time-consuming installation to train cars or maintenance vehicles.

An optical encoder (Figure 7 - Optical Encoder (Attached to Rail Wheel)) is used to trigger image capture and images are sent to a frame grabber to be digitized and then processed by the CPU. Image compression is performed on-the-fly using lossless algorithms to minimize data storage without compromising the usefulness of the data.

Figure 7 - Optical Encoder (Attached to Rail Wheel)

The LRAIL captures both Intensity and Range images of railway assets simultaneously. Intensity images (Figure 8 - Intensity Image) are produced by mapping the intensity of the reflected laser light, and Range images (Figure 9 - Range Image) are produced by mapping the elevation of each measurement point.

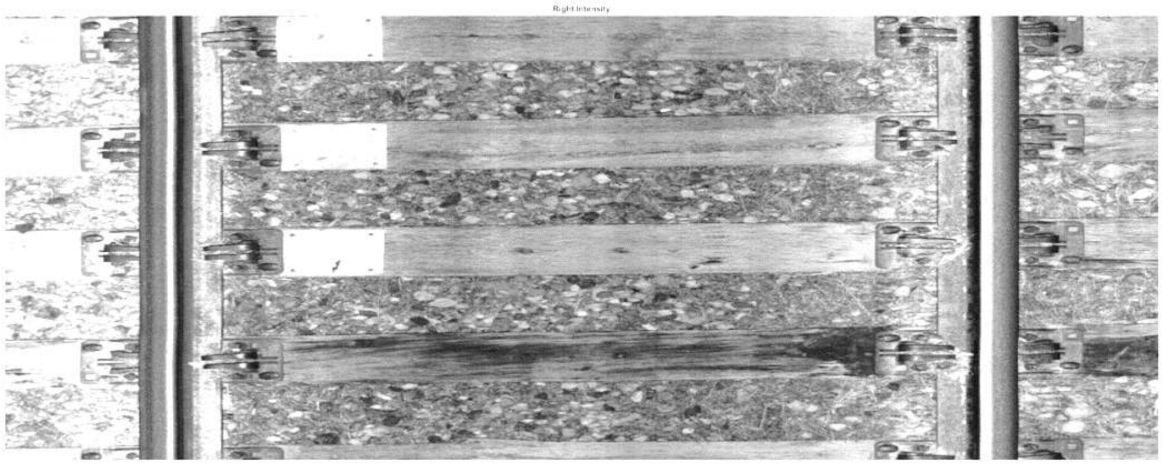

Figure 8 - Intensity Image

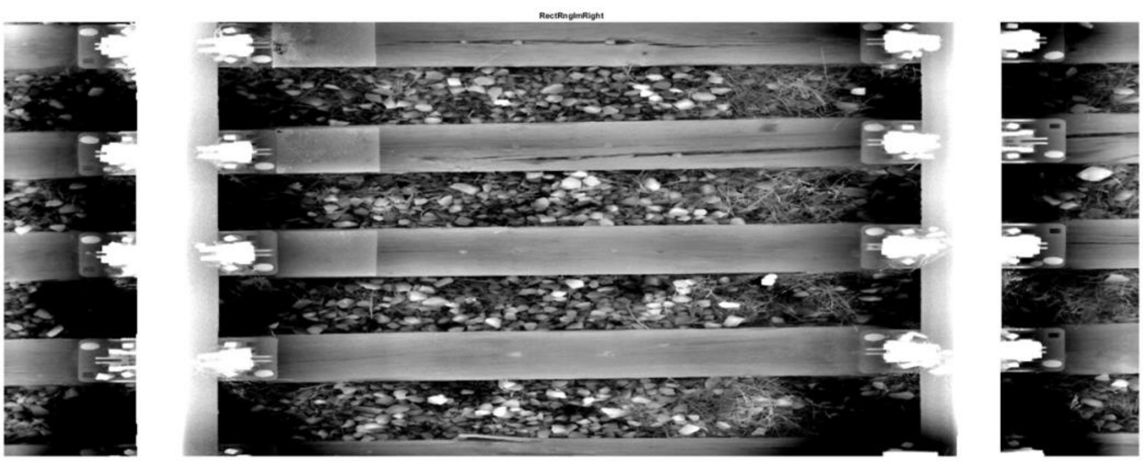

Figure 9 - Range Image

Resulting images are approximately 4,000 pixels wide with a lateral resolution of a point per millimeter and a longitudinal scan interval (as the inspection vehicle travels along the track) of a scan per millimeter. The end result is a 1 millimeter by 1 millimeter scan of the railway (Figure 10 - 1mm by 1mm Scan of Railway).

Figure 10 - 1mm by 1mm Scan of Railway

The key parameters of the LRAIL are presented in Table 1 - Key LRAIL Parameters:

Table 1 - Key LRAIL Parameters

Criteria	**Performance**
Longitudinal Scan Interval	Adjustable; 1 millimeter as tested
Vertical Resolution	0.1 millimeter
Horizontal Resolution	1 point per millimeter
Transversal Field of View	Adjustable; 3.5m as tested
Data Storage Requirements	Approximately 5GB per mile at 1mm scan intervals
Output Formats	JPEG Images, XML, LAS

3.4.1 Motion Compensation

As the high-rail vehicle traverses the track there is a great deal of vehicle vibration due to track conditions and features which would otherwise create distortions in 2D and 3D data preventing subsequent analysis. To overcome this challenge inertial sensors are incorporated into each laser head in order to track pitch, roll and heading (inertial) for each sensor while the inspection vehicle is in motion. These data are processed by specialized algorithms in order to remove their effect on captured scans (Figure 11 - Correction for Vehicle Motion).

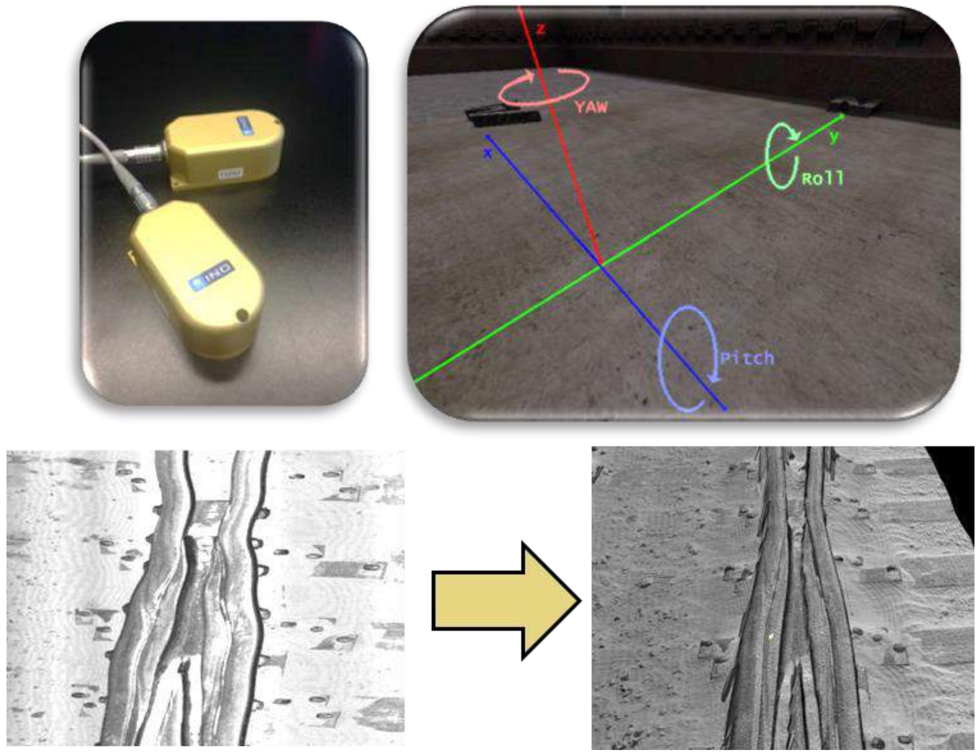

Figure 11 - Correction for Vehicle Motion

3.5 Image Classification Algorithm Details (Prior Work)

The image classification algorithm developed in prior work is a seven layer (6 hidden layers and 1 output layer) Supervised Machine Learning (SML) algorithm based on DeepCNet which combines 62,040 weights and 76 biases for a total of 62,116 learnable parameters (Figure 12 - DNN Diagram).

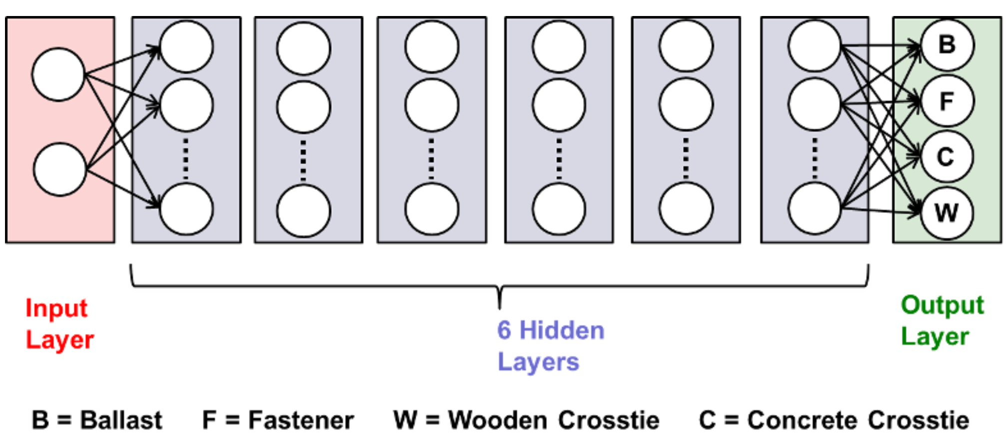

Figure 12 - DNN Diagram

The six hidden layers include: a 5x5 convolution layer with 5 maps, a 3x3 semi-stochastic pooling layer, a 1x1 inception layer, a 5x5 convolution with 10 maps, and a 2x2 semi-stochastic pooling layer that outputs to a fully connected output layer using softmax.

In order to determine if an image contains a fastener, some ballast, a concrete tie or a wooden tie, the DNN first divides each 4,000 pixel x 2,000 pixel image into two 35 x 35 pixel sub-images. Each 35 x 35 pixel image (increased to 151 x 151 in present work) is then labeled by the algorithm as either "fastener," "ballast," "wood," or "concrete" thus creating a "raw labeled" image.

Finally, the DNN groups sub images together in order to form regions which correspond to identified railway components (Figure 13 - Illustration of Processing Steps). This is used to eliminate small errors in sub image identification.

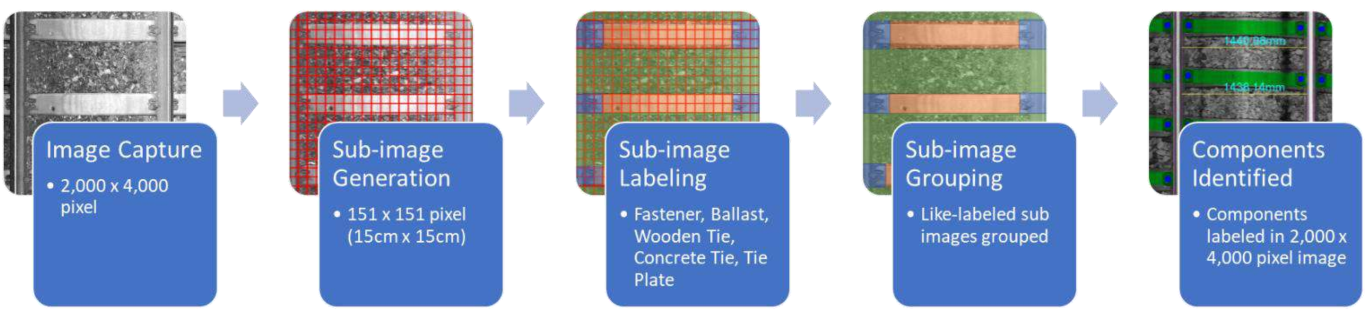

Figure 13 - Illustration of Processing Steps

3.6 Tie Plate Identification and Acceptability Rating Algorithm

Building on previous work, the DNN's Image Classification Algorithm was further trained to permit it to also identify tie plates (Figure 14 - DNN Identification of a Tie Plate) and a second phase of analysis was introduced in order to rate their acceptability (Figure 15 – DNN Tie Plate Rating).

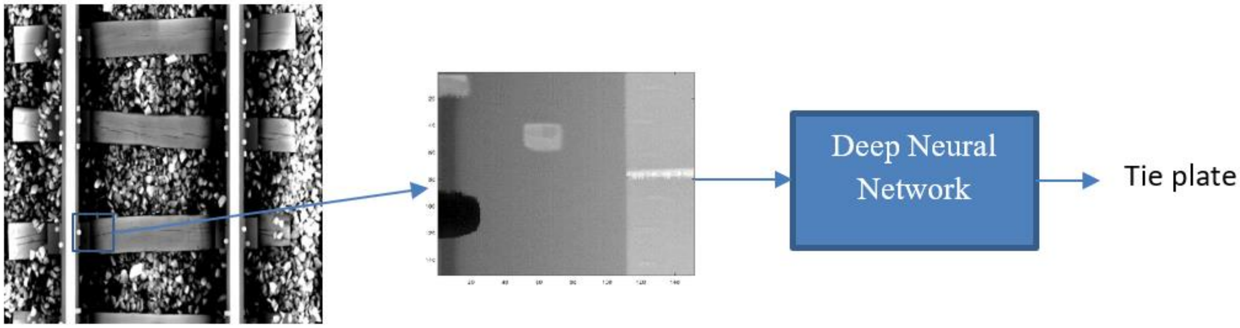

Figure 14 - DNN Identification of a Tie Plate

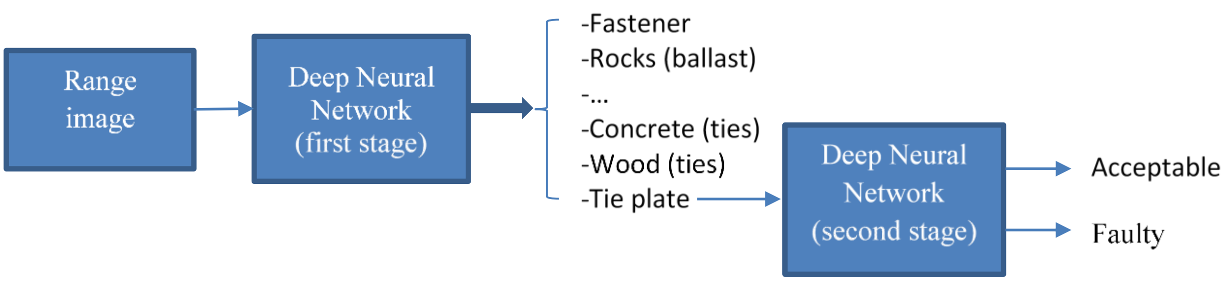

Figure 15 – DNN Tie Plate Rating

Tie plate acceptability was determined on the basis of the position and orientation of detected tie plates in relation to the rail; tie plates which were twisted, shifted to the left or right of the rail or sunken were considered faulty. Figure 16 - Acceptable Tie Plate (A) and Faulty Tie Plate (B)presents an example acceptable tie plate (A) that is properly positioned with the base of the rail sitting in its central channel compared to a faulty plate (B) where the tie plate is shifted to the right and the base of the rail is no longer resting in the middle of the plate channel.

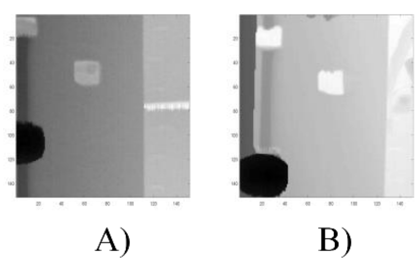

Figure 16 - Acceptable Tie Plate (A) and Faulty Tie Plate (B)

Figure 17 presents 3D profiles of an Acceptable (left) and a Faulty (right) Tie Plate; note that the Faulty Tie Plate is both shifted laterally away from the foot of the rail and is sunken along the vertical dimension.

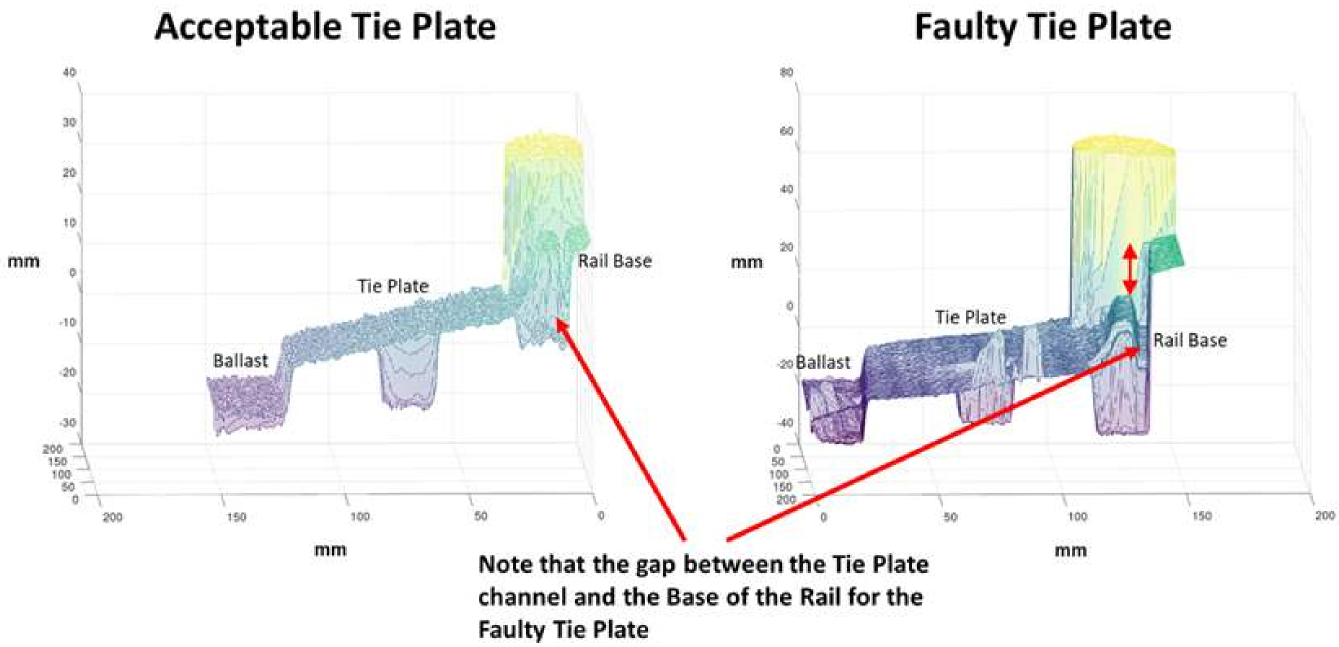

Figure 17 - 3D Scans of Acceptable vs. Faulty Tie Plate

3.7 Image Classification Training Process

The training of SML algorithms is iterative in nature with training data fed into the algorithm, the algorithm performing a classification, and finally feedback as necessary from a human "trainer" in order to improve performance over time (Brownlee, 2017) and (Marr, Supervised V Unsupervised Machine Learning -- What's The Difference?, 2017). The training process continues until no further improvements can be realized in classification performance; at this point the DL algorithm is said to have *converged*. Each complete run through of the training dataset is referred to as an *epoch* with a typical training cycle requiring fifty (50) or more epochs.

For this second project phase, both the training and testing datasets for Railway Component Identification were increased significantly to a total of 2,290 training images and

2,292 testing images (up from 200 for each set previously) for ballast, wooden ties, concrete ties and fasteners (1,542 contained fasteners, 639 wooden ties, 1,542 concrete ties, 771 ballast and 488 tie plate). For the new task of tie plate identification, a total of 1,550 full-size images were utilized.

Additionally, sub image size was increased from the original 35 x 35 pixel size to 151 x 151 pixels in order to increase the information available to the DNN. This resulted in longer training times but better overall performance.

During training sub-images containing example fasteners, ballast materials, concrete ties, wooden ties and tie plates were input into the DNN in order to generate a candidate image classification (Figure 18 - Fastener Classification Example).

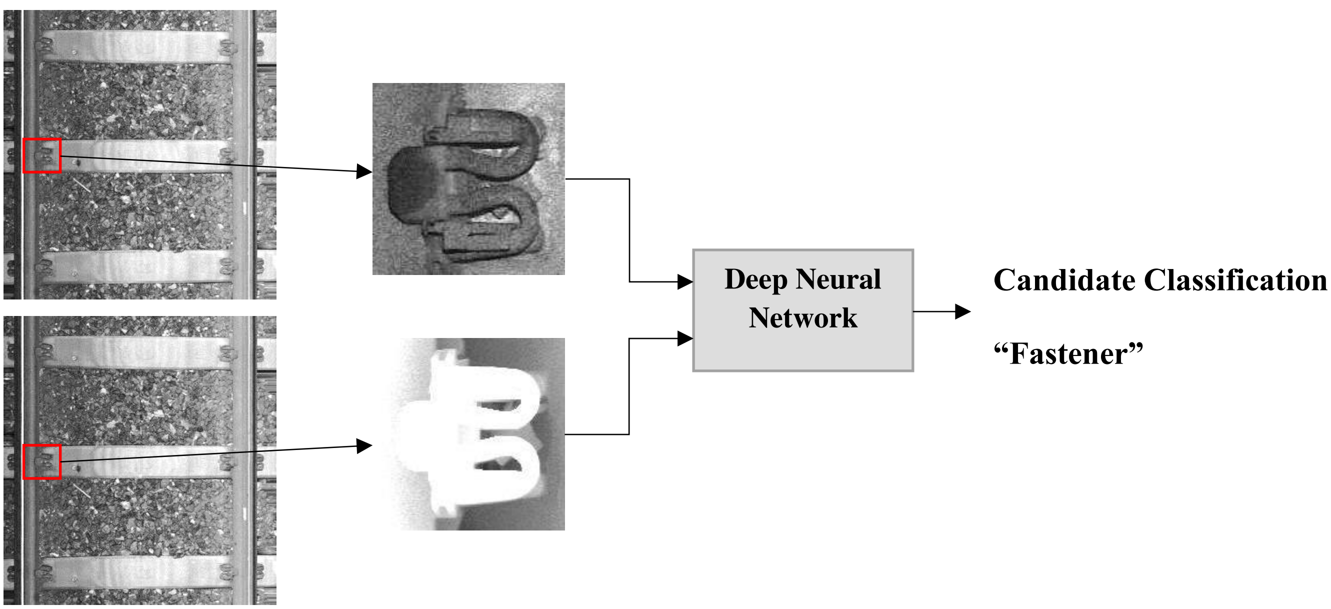

Figure 18 - Fastener Classification Example

Candidate classifications where then compared to the recommended classifications from the trainer; creating a feedback loop in order for the DNN to improve its recommendations. This training process was repeated a total of fifty times (50 epochs) and the DNN error (the difference between expected and actual classification) was calculated for each epoch in order to subsequently adjust the weight of algorithm parameters.

3.8 Tie Plate Rating Algorithm Training Process

For the Tie Plate Rating algorithm, a total of 2,852 images of acceptable tie plates and 82 images of faulty tie plates (twisted, shifted or hanging) were used for training and a matching set for testing. While it would have been ideal to utilize similar numbers of acceptable and faulty tie plates for training and testing purposes, this was not practical given the normal rate of occurrence of faulty tie plates for properly maintained track.

3.9 Algorithm Performance Testing

Once the training process was completed, the DNN was applied to the test dataset which consisted of images not prior presented to the DNN during training in order to evaluate performance using novel cases.

The performance of the Component Identification Algorithm was excellent; errors in the correct identification of components were below 1% for all component types (Table 2 – Component Identification Algorithm Performance).

Table 2 – Component Identification Algorithm Performance

	Fasteners	**Ballast**	**Concrete Ties**	**Wooden Ties**	**Tie Plates**
Number of Samples (Images)	1,542	771	1,542	639	488
Number of Errors in Component Identification	1	4	2	5	3
Percentage Error	0.06%	0.52%	0.13%	0.78%	0.61%

These small errors were then automatically eliminated by the DNN’s process of grouping like-labeled sub images (two images identified as containing a fastener for example) in order to identify regions containing the complete track component (fastener, tie, ballast, tie plate).

The performance of the Plate Rating Algorithm for Acceptable Tie Plates was impressive with an error rate of just above 1%, while the error rate for detecting faulty Tie Plates was slightly higher but still excellent at 2.44% (Table 3 - Error Rate for Tie Plate Rating Algorithm).

Table 3 - Error Rate for Tie Plate Rating Algorithm

	Acceptable Tie Plates	**Faulty Tie Plates**
Number of Samples (Images)	2,887	82
Number of Errors Made in Tie Plate Rating	32	2
Percentage Error	1.11%	2.44%

The implication of this performance is that the DNN was able to correctly identify an Acceptable Tie plate 99% of the time with just 1% of Acceptable Tie Plates being incorrectly flagged as Faulty (a 1% false positive rate). The implication for Faulty Tie Plates is that the algorithm was only able to find 97.6% of the faulty tie plates, and missed 2.4% of the faulty plates (a false negative rate of 2.4%).

4 Conclusion

This study builds on the success of prior work by increasing the level of training and testing for the railway component identification algorithm. It also adds the capability to identify tie plates.

The performance of the improved DNN Component Identification Algorithm was found to be excellent with an error rate less than 1% across all component types. Importantly even this small error is filtered-out by the DNN by post-processing of images to perform region grouping.

The performance of the Tie Plate Rating algorithm was also found to be excellent with nearly 99% of Acceptable Tie Plates and 98% of Faulty Tie Plates being correctly identified. The implication is that on a network-level inspection for Faulty Tie Plates relatively few false positives would be raised and the vast majority of Faulty Tie Plates would be detected by the DNN. Consequently, the authors feel that these initial results show a lot of promise for the ability of the DNN to not only correctly identify components but to also flag faulty ones. Future work includes expanded field work to capture additional training and testing image datasets, further training of the DNN to identify new components and training of the DNN to expand its condition rating to new components.

One interesting observation from the researchers is that much like every human brain, each DNN is unique due to its specific combination of nodes, layers and connections as well as the constant evolution of these same factors. This quality makes it impossible to define the exact methodology of any given DNN and has the interesting implication of preventing a DNN from being patented as well as preventing a DNN from infringing upon an existing patent.

References

Alex Chen, J. B. (2014, 02 10). *Distributed Neural Networks with GPUs in the AWS Cloud*. Retrieved from The Netflix Tech Blog: https://medium.com/netflix-techblog/distributed-neural-networks-with-gpus-in-the-aws-cloud-ccf71e82056b

Brownlee, J. (2017, 3 16). *Supervised and Unsupervised machine Learning Algorithms.* Retrieved from Machine Learning Mastery: https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/

Ciocca, S. (2017, 10 10). *Spotify's Discover Weekly: How machine learning finds your new music*. Retrieved from Hackernoon: https://hackernoon.com/spotifys-discover-weekly-how-machine-learning-finds-your-new-music-19a41ab76efe

Dayong Wang, A. K. (2016, 06 18). *Deep Learning for Identifying Metastatic Breast Cancer.* Retrieved from Cornell University Library: https://arxiv.org/pdf/1606.05718

John McCarthy (computer scientist). (2018, 2 14). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/John_McCarthy_(computer_scientist)

Kontzer, T. (2016, 09 19). *Deep Learning Drops Error Rate for Breast Cancer Diagnoses by 85%*. Retrieved from Nvidia Blog: https://blogs.nvidia.com/blog/2016/09/19/deep-learning-breast-cancer-diagnosis/

Laser Rail Inspection System (LRAIL). (2018, 02 14). Retrieved from Pavemetrics.com: http://www.pavemetrics.com/applications/rail-inspection/laser-rail-inspection-system/

Marr, B. (2016, 12 29). *4 Mind-Blowing Ways Facebook Uses Artificial Intelligence*. Retrieved from Forbes.com: https://www.forbes.com/sites/bernardmarr/2016/12/29/4-amazing-ways-facebook-uses-deep-learning-to-learn-everything-about-you/#204b273accbf

Marr, B. (2017, 3 16). *Supervised V Unsupervised Machine Learning -- What's The Difference?* Retrieved from Forbes.com: https://www.forbes.com/sites/bernardmarr/2017/03/16/supervised-v-unsupervised-machine-learning-whats-the-difference/#5fcd94a3485d

The story of AlphaGo so far. (2018, 02 14). Retrieved from AlphaGo: https://deepmind.com/research/alphago/