U.S. Department
of Transportation
Federal Railroad
Administration

Office of Research,
Development and Technology
Washington, DC 20590

Automated Train Operations (ATO) Sensor Platform Data Analysis Rapid Prototype

DOT/FRA/ORD-24-07

Final Report | February 2024

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof. Any opinions, findings and conclusions, or recommendations expressed in this material do not necessarily reflect the views or policies of the United States Government, nor does mention of trade names, commercial products, or organizations imply endorsement by the United States Government. The United States Government assumes no liability for the content or use of the material contained in this document.

NOTICE

The United States Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the objective of this report.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY*
09/30/2022

2. REPORT TYPE
Technical Report

3. DATES COVERED *(From - To)*
08/02/2021 – 09/30/2022

4. TITLE AND SUBTITLE
Automated Train Operations (ATO) Sensor Platform Data Analysis Rapid Prototype

5a. CONTRACT NUMBER
DTFR5311D00008L

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Martin Dvorak – ORCID: 0000-0003-4142-6852
Zach Vencius - ORCID: 0000-0001-6684-8053
Brian Helfin - ORCID: 0000-0003-0726-8029
Andrew Kelley - ORCID: 0000-0002-4555-6702

5d. PROJECT NUMBER

5e. TASK NUMBER
693JJ621F000036

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Transportation Technology Center, Inc.
55500 DOT Road
PO BOX 11130
Pueblo, CO 81001-0130

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Department of Transportation
Federal Railroad Administration
Office of Railroad Policy and Development
Office of Research, Development, and Technology (RD&T)
Washington, DC 20590

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
DOT/FRA/ORD-24-07

12. DISTRIBUTION/AVAILABILITY STATEMENT
This document is available to the public through the FRA Web site at http://www.fra.dot.gov

13. SUPPLEMENTARY NOTES
COR: Francesco Bedini

14. ABSTRACT

In a Federal Railroad Administration-funded Automated Train Operations (ATO) Sensor Platform (SP) Data Analysis Rapid Prototype (RP) project, a team from Transportation Technology Center, Inc. researched the feasibility of using commercial off-the-shelf (COTS) data analysis processes and tools to meet the needs of SP-related train automation functions defined in the ATO program. This effort, started in August 2021 and concluded in September 2022, primarily focused on evaluating the suitability of COTS algorithms for use in the analysis of data produced in a prior ATO SP RP project. This project included COTS algorithm identification, algorithm evaluation, and data analysis tasks.

15. SUBJECT TERMS
ATO, SP, Automated, Train, Operation, Sensor, Platform, Rapid, Prototype, Data, Analysis

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Martin Dvorak, Senior Engineer II
Unclassified	Unclassified	Unclassified		58	**19b. TELEPHONE NUMBER** *(Include area code)* 719-585-1824

METRIC/ENGLISH CONVERSION FACTORS

ENGLISH TO METRIC

LENGTH (APPROXIMATE)

1 inch (in)	=	2.5 centimeters (cm)
1 foot (ft)	=	30 centimeters (cm)
1 yard (yd)	=	0.9 meter (m)
1 mile (mi)	=	1.6 kilometers (km)

AREA (APPROXIMATE)

1 square inch (sq in, in^2)	=	6.5 square centimeters (cm^2)
1 square foot (sq ft, ft^2)	=	0.09 square meter (m^2)
1 square yard (sq yd, yd^2)	=	0.8 square meter (m^2)
1 square mile (sq mi, mi^2)	=	2.6 square kilometers (km^2)
1 acre = 0.4 hectare (he)	=	4,000 square meters (m^2)

MASS - WEIGHT (APPROXIMATE)

1 ounce (oz)	=	28 grams (gm)
1 pound (lb)	=	0.45 kilogram (kg)
1 short ton = 2,000 pounds (lb)	=	0.9 tonne (t)

VOLUME (APPROXIMATE)

1 teaspoon (tsp)	=	5 milliliters (ml)
1 tablespoon (tbsp)	=	15 milliliters (ml)
1 fluid ounce (fl oz)	=	30 milliliters (ml)
1 cup (c)	=	0.24 liter (l)
1 pint (pt)	=	0.47 liter (l)
1 quart (qt)	=	0.96 liter (l)
1 gallon (gal)	=	3.8 liters (l)
1 cubic foot (cu ft, ft^3)	=	0.03 cubic meter (m^3)
1 cubic yard (cu yd, yd^3)	=	0.76 cubic meter (m^3)

TEMPERATURE (EXACT)

[(x-32)(5/9)] □F = y □C

METRIC TO ENGLISH

LENGTH (APPROXIMATE)

1 millimeter (mm)	=	0.04 inch (in)
1 centimeter (cm)	=	0.4 inch (in)
1 meter (m)	=	3.3 feet (ft)
1 meter (m)	=	1.1 yards (yd)
1 kilometer (km)	=	0.6 mile (mi)

AREA (APPROXIMATE)

1 square centimeter (cm^2)	=	0.16 square inch (sq in, in^2)
1 square meter (m^2)	=	1.2 square yards (sq yd, yd^2)
1 square kilometer (km^2)	=	0.4 square mile (sq mi, mi^2)
10,000 square meters (m^2)	=	1 hectare (ha) = 2.5 acres

MASS - WEIGHT (APPROXIMATE)

1 gram (gm)	=	0.036 ounce (oz)
1 kilogram (kg)	=	2.2 pounds (lb)
1 tonne (t)	=	1,000 kilograms (kg)
	=	1.1 short tons

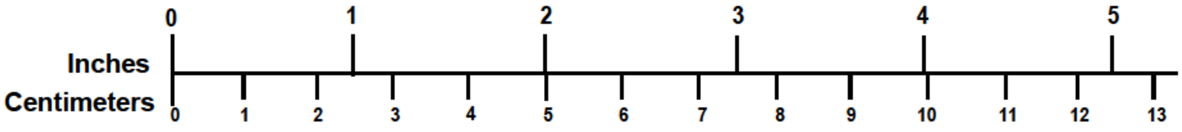
VOLUME (APPROXIMATE)

1 milliliter (ml)	=	0.03 fluid ounce (fl oz)
1 liter (l)	=	2.1 pints (pt)
1 liter (l)	=	1.06 quarts (qt)
1 liter (l)	=	0.26 gallon (gal)
1 cubic meter (m^3)	=	36 cubic feet (cu ft, ft^3)
1 cubic meter (m^3)	=	1.3 cubic yards (cu yd, yd^3)

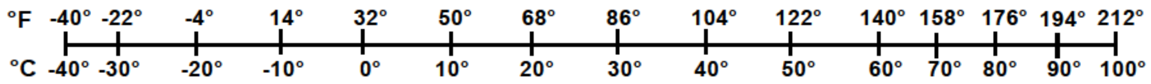
TEMPERATURE (EXACT)

[(9/5) y + 32] □C = x □F

QUICK INCH - CENTIMETER LENGTH CONVERSION



QUICK FAHRENHEIT - CELSIUS TEMPERATURE CONVERSIO


°F -40° -22° -4° 14° 32° 50° 68° 86° 104° 122° 140° 158° 176° 194° 212°
°C -40° -30° -20° -10° 0° 10° 20° 30° 40° 50° 60° 70° 80° 90° 100°

For more exact and or other conversion factors, see NIST Miscellaneous Publication 286, Units of Weights and Measures. Price $2.50 SD Catalog No. C13 10286

Updated 6/17/98

Contents

Illustrations

Tables

Executive Summary

As part of a Federal Railroad Administration (FRA)-funded Automated Train Operations (ATO) Sensor Platform (SP) Data Analysis Rapid Prototype (RP) project, a research team from MxV Rail, (formerly Transportation Technology Center, Inc.), evaluated commercial off-the-shelf (COTS) data processing techniques with the potential to support ATO locomotive SP-related train automation functions.

Current industry efforts, as well as several FRA-funded efforts, are underway to develop ATO for use in North American Class I freight railroad service. ATO, coupled with existing safety and efficiency enhancing systems (e.g., Interoperable Train Control (ITC) Positive Train Control (PTC) and train energy management systems (EMS)) offer the North American rail industry safety and operational enhancements, allowing it to remain a competitive and viable long-haul transportation mode. The objective of the ATO SP Data Analysis RP project, conducted by TTCI from August 2021 to September 2022, was to study the capability of COTS data processing techniques to support SP functions. This project was a continuation of the ATO SP RP project and continued the data analysis work started in the prior project.

The team investigated clear path detection and person detection using sample data sets and railroad-specific data sets. Researchers evaluated autoencoder and saliency-based approaches for clear path detection. Further refinement of these techniques may provide a hazard detection solution without the need for producing an algorithm capable of classifying every possible hazard.

In the person detection effort, the team evaluated 10 different neural networks. Researchers found that while the performance of each could be improved, no single neural network appeared capable of producing the results necessary for an SP. However, the team found substantial improvement by using a committee of neural networks to eliminate false negatives. These results suggest the potential of meeting reliability requirements using multiple neural networks trained on the specific hazards found in the rail environment.

Future work should include the collection of larger data sets involving the rail environment, proper annotation of those data sets for use in training and evaluating machine learning algorithms, improving the performance of the algorithms studied, evaluation of additional algorithms, and improved committee algorithms.

1. Introduction

In a Federal Railroad Administration (FRA)-funded Automated Train Operations (ATO) Sensor Platform (SP) Data Analysis Rapid Prototype (RP) project, a team from Transportation Technology Center, Inc. (TTCI) researched the feasibility of using commercial off-the-shelf (COTS) data analysis processes and tools to meet the needs of SP-related train automation functions defined in the ATO program. This effort, started in August 2021 and concluded in September 2022, primarily focused on evaluating the suitability of COTS algorithms for use in the analysis of data produced in a prior ATO SP RP project. This project included COTS algorithm identification, algorithm evaluation, and data analysis tasks.

1.1 Background

The railroad industry is engaged in an ongoing effort to define an interoperable ATO system of systems. ATO encompasses a collection of train automation functions that individually provide improvements to railroad efficiency and safety and collectively provide for high automation for trains under normal line of road operating conditions. The capability of machine perception of possible hazards within the railroad operating environment is a potentially enabling technology for many of the train automation functions that are expected to enhance railroad safety and efficiency. These train automation functions are intended to be fully interoperable, allowing any equipped train to seamlessly operate across any ATO-equipped North American railroad and supported by any qualified railroad personnel, regardless of the automation equipment supplier.

Within the ATO concept, the SP is responsible for scanning the external environment ahead of the locomotive and providing actionable information to locomotive onboard systems. An SP supporting high automation is expected to monitor the foul volume and right of way ahead of the train and provide high confidence information regarding the distance to which the train route is clear, classification of Objects of Interest (OOIs) that may be fouling the track or occupying the right of way, and Conditions of Interest (COIs) in the roadbed and right of way that present a hazard. An SP supporting a limited set of train automation functions may perform a specified subset of the functionality performed by the full SP. Train automation systems consuming SP-provided information are responsible for initiating appropriate train responses as governed by railroad operating practices and regulatory requirements.

In a prior FRA-sponsored ATO Safety and Sensor Development project (Federal Railroad Administration, 2020) researchers defined interoperable requirements for a SP capable of supporting ATO. These requirements defined the OOIs and COIs to be detected and the regions in which they are to be detected and started identifying the reliability performance of the detection system.

The ATO Safety and Sensor Development project was followed by the FRA-sponsored ATO SP RP project, in which researchers began evaluating the feasibility of constructing an SP using COTS sensor devices. The team built a sensor array, collected data, and performed preliminary analysis. This report details the further analysis of this data to study the feasibility of using modern data analysis software to meet the SP requirements.

1.2 Objectives

The ATO SP Data Analysis RP effort included the following objectives:

- Demonstrate the capability of COTS data processing techniques to perform SP functions such as object detection, object classification, and anomaly detection
- Provide advisement and modification of sensor platform requirements associated with data analysis function and performance
- To the extent possible with available railroad-provided data, compare the SP RP system to current operations

1.3 Overall Approach

The team performed project management and engineering tasks in collaboration with railroad industry representatives. As part of the broader ATO development effort, a Technical Working Group (TWG) comprised of railroad and FRA members provided technical input and oversight to the team's ATO technical efforts. This TWG served as the advisory group (AG) for the ATO SP RP project. As they were already familiar with the SP RP effort, this group then served as the AG for this project, providing guidance on project goals and technical oversight. To assist in the data analysis system evaluation efforts, the team also contracted with a sensor systems consultant.

The team executed the project in three phases, as illustrated in Figure 1:

- Identify COTS tools
- Implement data analysis processes
- Analyze effectiveness of analysis processes

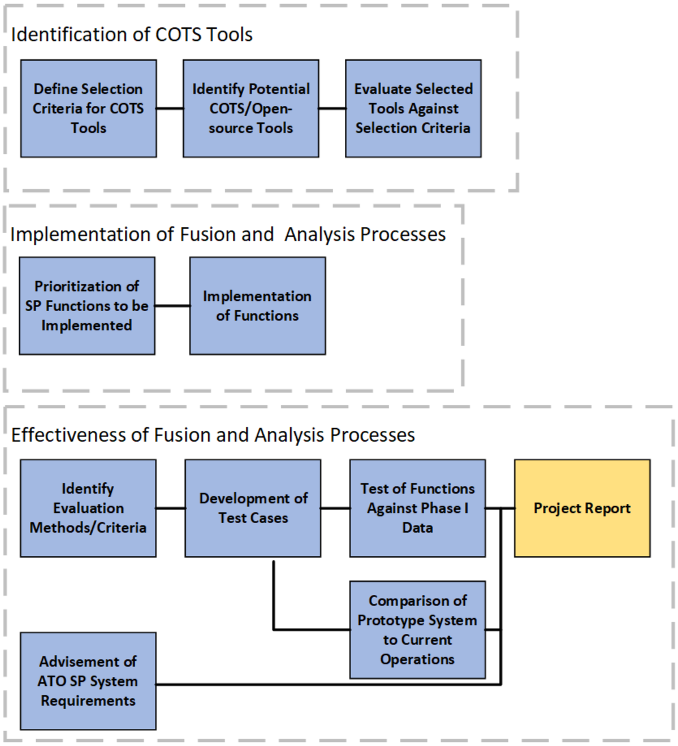

Identification of COTS Tools
Define Selection Criteria for COTS Tools
Identify Potential COTS/Open-source Tools
Evaluate Selected Tools Against Selection Criteria
Implementation of Fusion and Analysis Processes
Prioritization of SP Functions to be Implemented
Implementation of Functions
Effectiveness of Fusion and Analysis Processes
Identify Evaluation Methods/Criteria
Development of Test Cases
Test of Functions Against Phase I Data
Project Report
Comparison of Prototype System to Current Operations
Advisement of ATO SP System Requirements

Figure 1. Project task flow diagram

1.3.1 Identify COTS Tools

During the initial project phase, researchers used a three step process to identify and select COTS data analysis tools:

- Define selection criteria for COTS tools
- Identify potential COTS and open-source tools
- Evaluate selected tools against selection criteria

The team needed to define the selection criteria for COTS data analysis tools before useful tools could be selected. This included problem definition, market research, and definition of selection criteria. Analysis goals included the detection, classification, ranging, tracking, and intercept prediction of objects.

Once selection criteria were defined, the team conducted a survey of COTS and open-source analysis tools to identify those potentially capable of meeting project goals. Researchers then evaluated the tools identified against the selection criteria, and the necessary tools procured.

1.3.2 Implement Data Analysis Processes

During the implementation phase, the team prioritized and implemented the SP functions. The top priority was the implementation of a clear path detection algorithm. The detection of people, vehicles, and other objects was a lesser priority.

1.3.3 Analyze Effectiveness

The team analyzed the effectiveness of the data analysis processes by:

- Identifying evaluation criteria
- Developing test cases
- Testing software against available data
- Comparing the results to current operations

In addition, the results of this project are being considered during revision of the existing SP system requirements outside the scope of this project.

1.4 Scope

The team performed the following tasks as part of the ATO SP Data Analysis RP project:

- Identified and procured COTS data analysis tools that have potential to satisfy ATO SP requirements
- Implemented data analysis processes to identify if the path ahead is clear, with detection and classification of objects as secondary objectives
- Applied data analysis processes to sensor data collected from the sensor suite assembled in the prior ATO SP RP project
- Analyzed the effectiveness of data analysis processes at performing their intended function
- Reported project findings and recommendations for next steps

The scope was limited to an initial evaluation of COTS tools, with development of custom data analysis processes and refinement of existing processes considered out of scope. The intent was to begin evaluating what capabilities may already be available to determine future efforts that will be required, and not to perform an exhaustive evaluation of all potential data analysis processes. Additionally, the scope of the evaluation was limited by the project budget and schedule, with priority given to evaluation of processes to determine if the path ahead of the train is clear.

1.5 Organization of the Report

This report summarizes and highlights the results of the ATO SP Data Analysis RP project. The report is organized as follows:

- Section 1 provides the objective and background information of the project to aid in setting the context of the project.
- Section 2 reviews sensor platform concepts and requirements.
- Section 3 provides an overview of clear path detection algorithms developed during this project using saliency and autoencoder neural network-based methods.
- Section 4 provides an overview of existing open-source neural networks and their use in object classification tasks, as well as an overview of an output fusion method and the impact of output fusion on performance.
- Section 5 provides the report conclusion and recommended next steps.

2. Sensor Platform Concept and Requirements

A prior FRA-sponsored ATO Safety and Sensor Development project (Federal Railroad Administration, 2020) informed about this effort. An SP capable of supporting train automation functions could use any of a wide range of COTS technologies that may include, cameras (optical, thermal, infrared), lidar, radar, sonar, or other technologies. Because the prior ATO SP RP project used camera technology, this research team focused on the analysis of data produced by cameras. To understand the data analysis objectives, it is necessary to understand the basic SP functionality and a small set of selected use cases.

The SP monitors the environment ahead of the train for external conditions that present hazards, such as track obstructions or people encroaching within the foul volume. The environment ahead of the train is logically partitioned into three areas of interest, the foul volume, the collision volume, and the wayside. These logical partitions are illustrated in Figure 2.

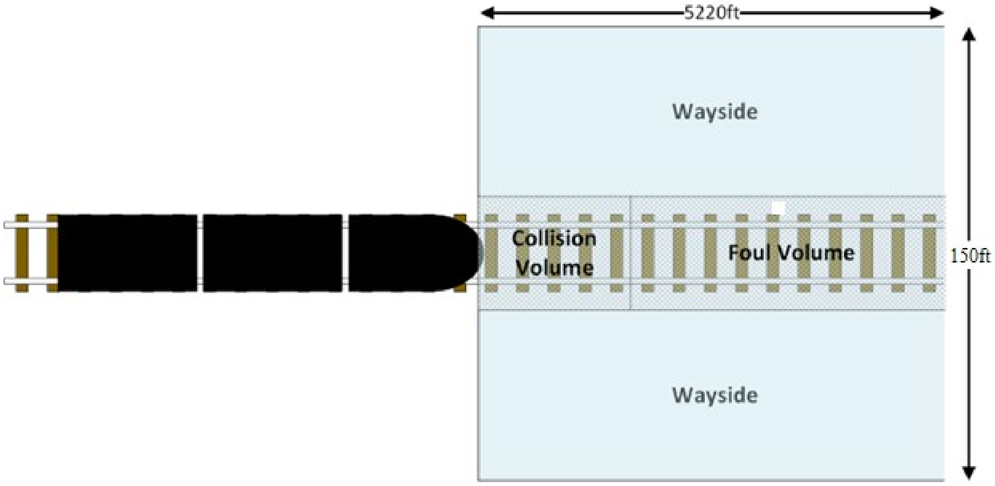


Figure 2. Areas of interest

External conditions are further defined as either:

- Objects of Interest (OOI) – Objects in an area of interest, not part of the railroad track infrastructure, that may pose a hazard to the train (e.g., people, vehicles, etc.), or
- Conditions of Interest (COI) – Objects/conditions in an area of interest that are part of or impacting the railroad track infrastructure that present a hazard to the train (e.g., sun kinks, earth over rail, failed crossing gates, etc.).

The foul volume is the region ahead of the train through which the train will pass. Definition of the foul volume depends on the AAR plate to which a track is built. For the purposes of this project, researchers defined the foul volume to be 10 feet wide and centered on the track centerline and extending from the surface of the rail head to 15 feet above the rail head. The foul volume extends along the train route from the end of the collision volume in front of the lead locomotive, to a distance defined by the SP use cases. An SP must be able to distinguish between an OOI located just inside the foul volume and just outside the foul. For this reason, a data analysis objective was the localization of an object relative to the track occupied by the train and the leading edge of the train.

The collision volume is the area immediately in front of the train. Any object within the collision volume is considered so close to the train that a collision is unavoidable. The final definition of the collision volume may take the speed of the train into account, but for this effort it was considered the region from the leading edge of the lead locomotive to 6 feet in front of the lead locomotive. The collision volume is centered on the track centerline and has the same width and height as the foul volume.

The wayside is the area to either side of the foul and collision volumes that may need to be monitored for OOIs and COIs. For this project, researchers defined the wayside as being within 70 feet of either side of the foul and collision volumes.

2.1 Sensor Platform Use Cases

The following SP use cases represent a subset of the full SP functionality as defined in prior efforts. These use cases were selected as a subset representative of SP functions, the investigation of which is of benefit to industry efforts to develop an interoperable SP. These SP use cases informed the data analysis objectives.

2.1.1 Clear Path

The core function of the SP is monitoring for environmental conditions (OOIs or COIs) which result in it being unsafe for a train to proceed. Current data analysis approaches focus on positively identifying known phenomena (e.g., inanimate objects, people, animals, or other conditions such as fire, rain, or fog). If a machine vision system does not identify any objects in a scene, it is assumed that nothing of interest is in the scene. This approach has both technical and practical benefits and supports the other use cases.

However, for this system to verify that it is safe for a train to proceed, it would require an exhaustive data set describing every possible phenomenon which could render it unsafe to proceed. This could be considered impractical for many reasons, including:

- Many potentially hazardous objects, such as rocks and downed trees, vary tremendously. Training datasets may fail to encompass every possible rock and downed tree that could obstruct a railroad track.
- Objects may present a different appearance while obstructing a railroad track than they do in normal training data sets. For example, cargo fallen off a train onto an adjacent track will have a highly variable appearance depending on the damage and final resting position.
- Objects unexpected in the rail environment may, on rare occasion, be found on the track. For example, in January 2022, an airplane crash-landed, coming to a stop on an active railroad track (National Transportation Safety Board, 2022).
- New vehicles will be found at grade crossings in the future that have not been designed yet; it is not possible to definitively characterize all of them.

One approach to addressing this challenge is clear path detection. A clear path can be defined as any condition an SP is to detect. Any obstruction preventing a clear path from being observed is detected regardless of the nature of the obstruction; an object not considered during algorithm design would be detected as the lack of a clear path. This includes any deviation of the rails from the definition of a clear path, regardless of the nature of the deviation. For example, gross rail damage as would be detected by a crew (e.g., sun kinks) could be detected regardless of the specifics of the sun kinks. Clear path detection differs from object detection in that the distance to which the path is clear must be measured. This requires an SP that, within acceptable confidence limits, reports the distance to which the foul volume is clear of all objects. Such an SP could then be expanded with definitions of known non-hazards, resulting in improved performance over time.

2.1.2 Person in Wayside or Foul

People in the wayside or foul commonly include:

- Railroad personnel
- The general public (frequently present at grade crossings)
- Trespassers

People could be present in any of the areas of interest and the exact location is important. A person standing just outside the foul volume must be distinguished from the same person stepping into the foul volume. In addition, people will commonly be found outside the areas of interest; they are clutter and should be disregarded.

Of special concern for the development of a functional SP is the ability to distinguish people from livestock and from people-like objects (e.g., a picture of a person printed on a sign adjacent to the track). Researchers considered this concern secondary for the initial data analysis effort, but it is likely to be a major factor in later efforts.

2.1.3 Vehicle in Wayside or Foul

Vehicles are commonly found at grade crossings and on roads alongside the track. As with people, the vehicle location is required. Size, composition, and appearance are expected to easily distinguish vehicles from other OOIs. Of more concern is distinguishing a vehicle from clutter. For example, a vehicle driving on a public road adjacent to the track is clutter, while a vehicle driving on a service road within the wayside is an OOI.

2.1.4 Livestock

Livestock are commonly found along some railroad tracks, depending on the nature of fencing along the track and proximity to ranching activity. As noted in Section 2.1.2, the primary concern is correctly distinguishing livestock from people.

2.1.5 Other Hazards

A wide range of other hazards may be encountered in the rail environment. Researchers did not attempt an exhaustive characterization of other hazards for this effort. Instead, the team considered the detection of objects with a profile of at least 1 foot by 1 foot as desirable. This informs the minimum size of OOI that the data analysis algorithms need to detect.

2.2 Detection, Localization, and Classification

The purpose of the data analysis is to identify all OOIs and COIs present in the foul volume and wayside. The SP will operate in a noisy, cluttered environment. Noise includes the presence of signals which may produce undesired or incorrect sensor readings. For example, the setting sun may shine into a camera, saturating the sensor and diminishing the quality of data from the sensor. Clutter includes the signal generated by real objects which are present, but not of interest. Trash alongside the track is clutter, as is everything outside the areas of interest. One data analysis goal is to differentiate real OOIs and COIs from noise and clutter.

2.2.1 Detection

Detection is the identification of the number of OOIs and COIs present. Detection characterizes OOIs and COIs only in that a decision is made as to if a detection is noise/clutter or potentially an OOI or COI. An object not of interest may be discarded as clutter during detection or reported as a detection and later discarded during classification. Detection does not classify OOIs, COIs, and objects not of interest beyond the decision to report them as detections. For example, two people in the wayside would be reported as two distinct detections; a washed-out track and a fuse would also be reported as two distinct detections.

The goal of the SP is to detect all OOIs and COIs present within the areas of interest. In practice, SP detection will be limited by the field of view (FOV) of the sensor devices. The FOV is limited both by environmental obstructions and the sensor technology used. An additional challenge is limiting detections to the areas of interest.

As the SP must be able to detect many OOIs and COIs at the same time, detection is closely linked to classification and localization. For an OOI or COI to be classified and localized, its presence must be detected, and that detection must correctly report separate OOIs and COIs as separate detections.

2.2.2 Localization

Localization is the process of finding the physical location of OOIs and COIs in the real world. In this case, OOIs and COIs are located relative to the head of the train and the foul volume. The SP will report the distance along the train route to the point on track closest to the OOI or COI and whether it is in the foul volume.

To support localization of OOIs and COIs, the sensor devices must produce data from which the real-world position can be calculated relative to the Head of Train (HOT). Localization approaches frequently involve sensors providing the azimuth angle, elevation angle, and distance relative to the sensor(s). However, other localization approaches can be used, such as stereoscopic ranging. Localization was addressed in the initial ATO SP RP project.

2.2.3 Classification

Classification is identifying the nature of an OOI or COI well enough to assign it to one of several pre-determined classes. Classification differs from full identification in that information beyond the object class is not needed. For example, a 1973 Dodge Viper is simply classified as a vehicle; the specific make and model do not matter. Some detections will be objects not of interest (clutter) or noise and are discarded. Detections which cannot be classified at the required confidence are reported as unknown OOIs.

3. Clear Path Detection

As discussed in Section 2.1.1, clear path detection is a priority approach intended to handle the highly variable nature of potential hazards. Due to this high variability, the team did not consider it possible to train a classification network with enough classes to account for every possible potential hazard. The team explored two primary clear-path detection methods in this project: saliency-based target detection and autoencoder neural networks.

3.1 Saliency Based Target Detection

Researchers considered the possibility of visual saliency-based target detection for use in clear path detection, locating objects ahead of the locomotive, and calculating the distance to the objects. This section introduces the visual saliency process and discusses two types of visual saliency: spectral residual static saliency and fine grain static saliency.

3.1.1 Visual Saliency

A visual saliency map, or saliency map, shows the region(s) of an image on which people's eyes tend to focus first, providing an indication of the importance of each pixel to the human visual system. Saliency maps are themselves an image, with the brightness of each pixel representing how salient the corresponding pixel is in the original image. They are normally presented as heat maps with bluer pixels representing lower saliency and redder pixels representing higher saliency. A shadow of the original image may be retained in the saliency map to allow easy human interpretation of the saliency map. The saliency maps used in machine vision are algorithmically produced and attempt to mimic the saliency of each area in the image as perceived by a human. Saliency maps are used in various visual attention models such as those from Itti & Coch (2001). Figure 3 provides an example of a visual saliency map showing the input image (left) with the computed saliency map (right).

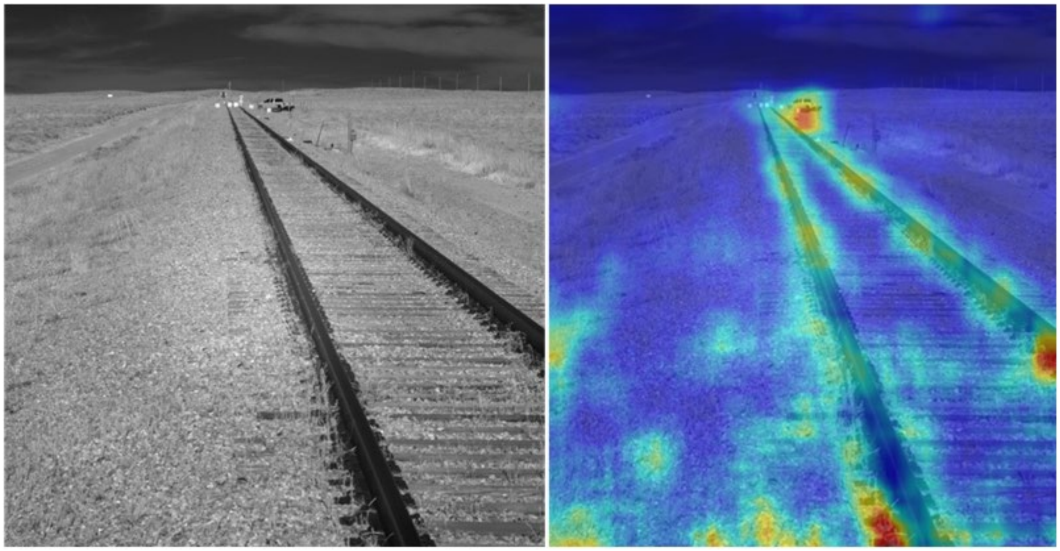


Figure 3. Input image (left) and computed saliency map (right)

In this effort, the focus was on static saliency, in which the features of a single image are analyzed for probable visual cues. Two static saliency detection algorithms are described in the following subsections.

3.1.1.1 Spectral Residual Static Saliency Detection Algorithm

The research team tested the log-spectrum based algorithm presented in Hou (2007). This algorithm is based on analyzing the log-spectrum of an input image and extracting the spectral residual. The log-spectrum is the graph of image intensity (on a logarithmic scale) versus the image frequency. In this context, image frequency refers to the rate of change at any point in the image. Smooth parts of an image with no sharp edges or sharp changes of contrast will have a low image frequency. Parts of an image with rapidly changing contrast and overlapping object edges will have a high image frequency. The residual spectrum is the result of subtracting the average log-spectrum (found by averaging the log-spectrum of multiple images within the image set) from the log-spectrum of the image of interest. The remaining values are the residual spectrum in the frequency domain, representing areas of the image that contain possible proto-objects, or objects which may be of higher interest or draw a viewer's focus. The frequency domain residual spectrum is transformed into the spatial domain to generate a saliency map. Figure 4 shows a spectral residual saliency map overlaid as a heatmap on the input image. The red and yellow region represents the salient areas in the image, with red as the most salient.

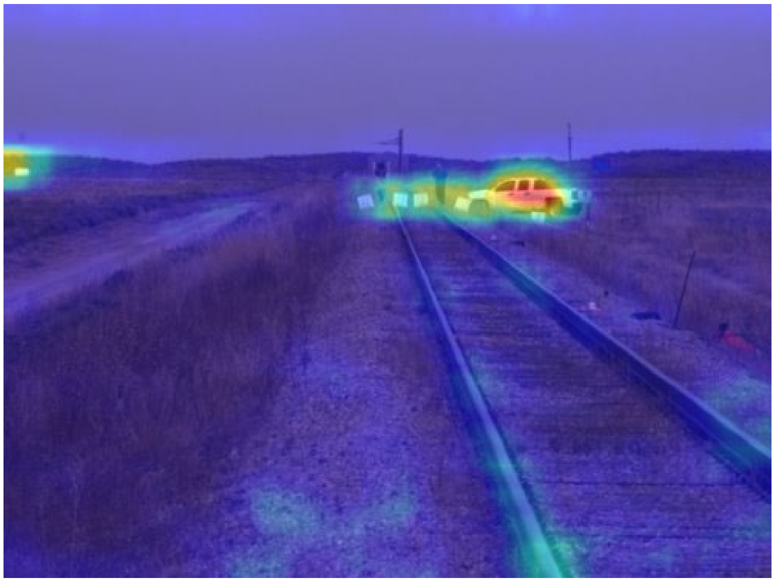


Figure 4. Spectral residual saliency map overlaid onto input image

3.1.1.2 Fine Grained Static Saliency Detection Algorithm

Montabone & Soto (2010) present a saliency algorithm that produces a fine-grained feature map of visual saliency by using an efficient implementation of center-surround differences through the so-called "integral image." Given an input image, the resulting pixels of an integral image are comprised of the sum of pixel values to the left of and above the input image pixels. Combining the integral image in various ways with the original input image can produce a fine-grained saliency map where the proto-object shape is better defined than in the spectral residual approach. This algorithm can operate in real time at the original image resolution. Figure 5 shows the original input image and the fine-grained saliency map overlaid as a heatmap onto the input image where the red and yellow region represents the most salient areas in the image. This approach shows an improvement over the spectral residual approach as the proto-object locations are far less diffused and better represent objects within the image.

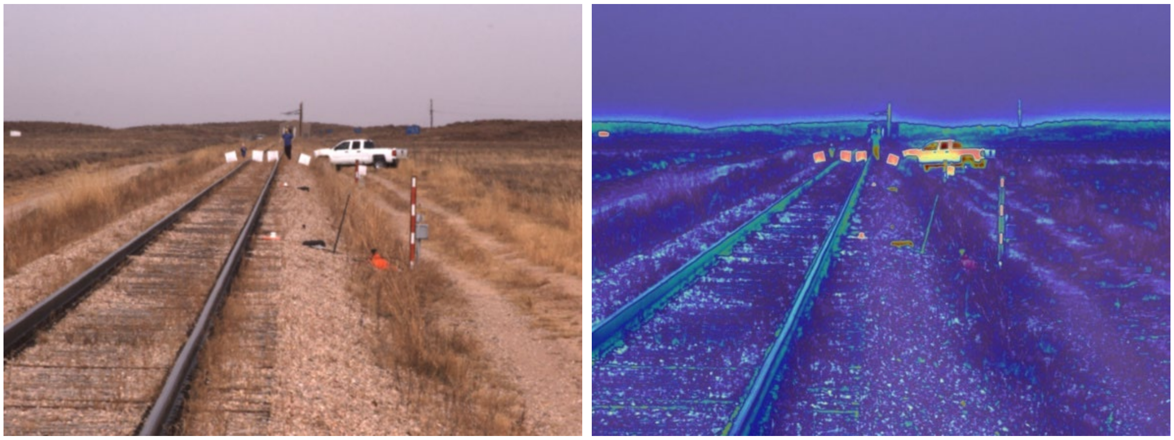


Figure 5. (left) Input image, (right) fine-grained saliency map

3.1.2 Thresholding

The second step of the clear path detection algorithm is to threshold the computed saliency map. The purpose of thresholding is to eliminate areas where the saliency value is too low and does not warrant further inspection. Additionally, the thresholded saliency map is the input to the segmentation algorithm presented in the next section. Thresholding comes in three forms: 1) simple thresholding where the user specifies a global threshold value to use, 2) Otsu's thresholding (Otsu, 1979) where the global threshold value is automatically determined, and 3) adaptive thresholding that will determine a threshold value for different pixel regions in the input image. This approach is useful for images that have different lighting conditions in different areas. For preliminary thresholding under this project, the team used simple thresholding. Figure 6 and Figure 7 show the thresholded saliency maps, using the input image shown in Figure 5, for the spectral residual and fine-grained method, respectively.

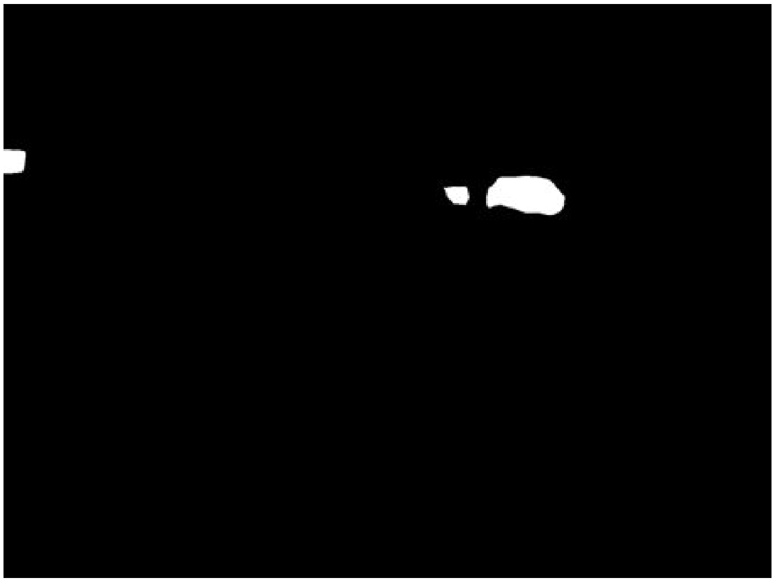


Figure 6. Spectral residual saliency map for input image in Figure 5

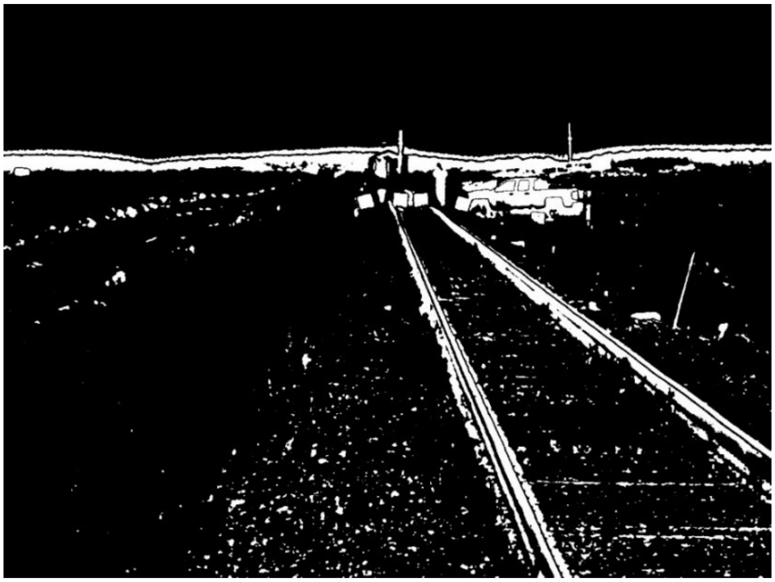


Figure 7. Fine grained saliency map for input image in Figure 5

3.1.3 GrabCut Based Image Segmentation

After thresholding of the saliency map, the original image and the output of the thresholding algorithm (Section 3.1.2) are processed by the GrabCut algorithm to extract the foreground. GrabCut is an image segmentation algorithm described by Carsten Rother (2004). In this case, the foreground is defined as the set of areas that may contain targets of interest, which is not necessarily the same area a photographer would define as the foreground. The background is then the image area that is not part of the foreground.

The GrabCut algorithm uses the thresholding map as the approximate segmentation, defines those areas as the foreground pixel set, and considers all the other areas as the background pixel set. Next, foreground and background Gaussian Mixture Models (GMMs) are created using the previously defined background and foreground pixel sets. An iterative process is then used to learn GMM parameters to create new pixel distributions. A graph cut optimization is then performed to reach the final image segmentation. The user will specify how many iterations of the process will be executed. Figure 8 and Figure 9 show the foreground extracted from the input image (Figure 5) using the GrabCut algorithm for the spectral residual and fine-grained method of saliency, respectively.

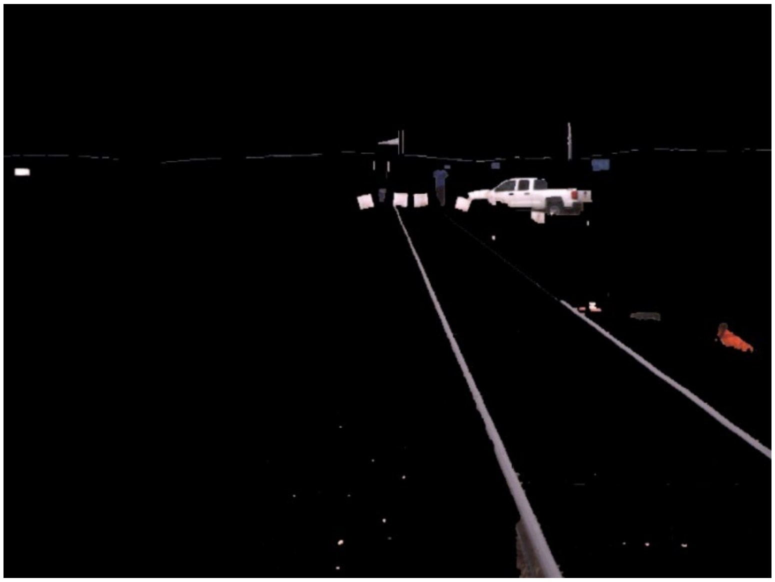


Figure 8. Foreground extracted from the input image in Figure 5, for the spectral method of saliency using the GrabCut algorithm

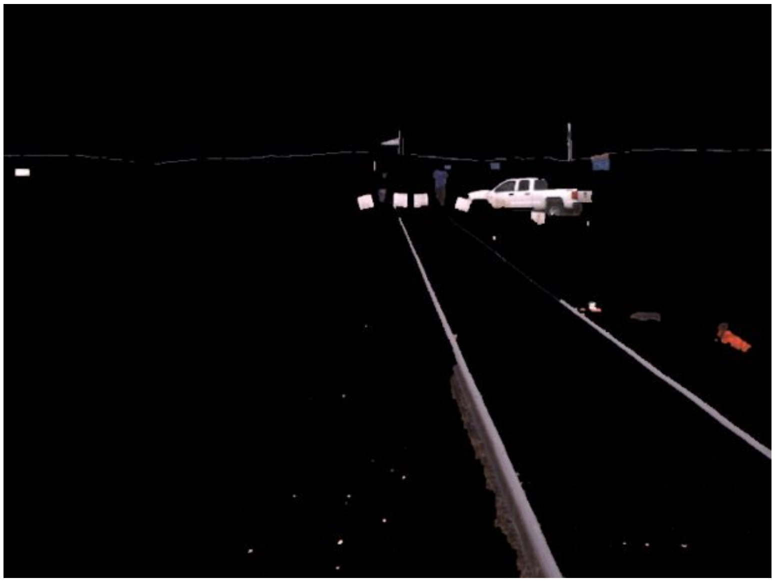


Figure 9. Foreground extracted from the input image in Figure 5, for fine-grained method of saliency using the GrabCut algorithm

3.1.4 GrabCut Threshold

The next step in the target detection process is to threshold the foreground output from the GrabCut algorithm. A simple threshold value of 1 is used to convert the input image to a binary image. Figure 10 and Figure 11 show examples of GrabCut binary image outputs for the spectral and fine-grained methods of saliency, respectively.

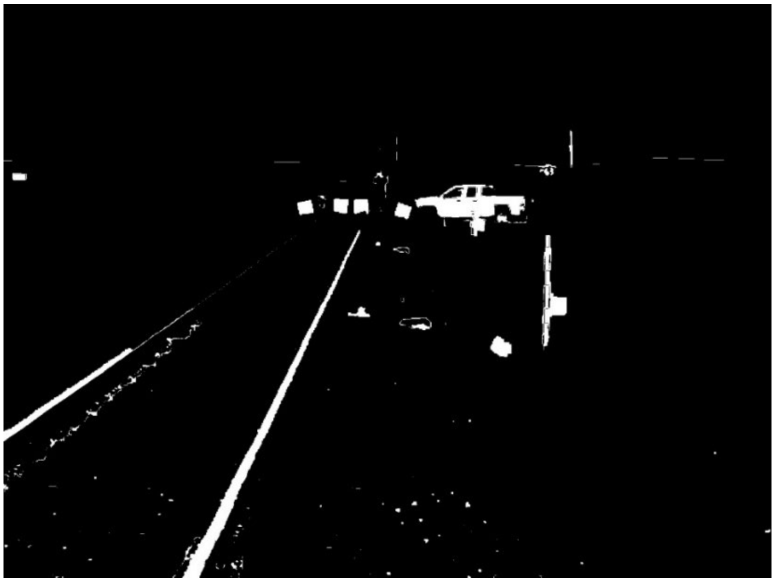


Figure 10. Binary image of the GrabCut output in Figure 8 for the spectral method of saliency

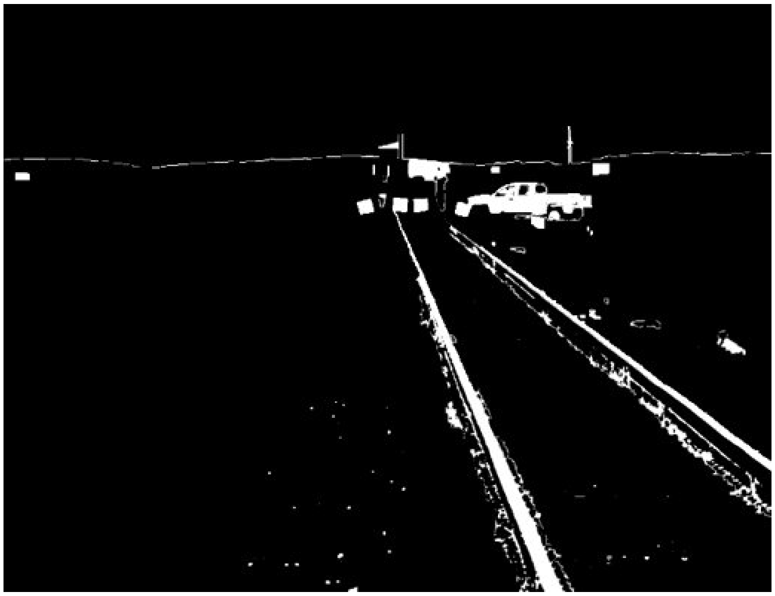


Figure 11. Binary image of the GrabCut output in Figure 9 for the fine-grained method of saliency

3.1.5 Object Localization

Object localization is used to describe both the process of locating an object in the real world relative to the head of the train, and to describe the process of locating an object within an image. Locating an object within an image supports locating the object in the real world, as explained below.

To localize objects in the thresholded GrabCut output, the OpenCV "findContours" method is first used. The findContours method, presented in Structural Analysis and Shape Descriptors (2022), uses the black and white image produced by the GrabCut Thresholding (Section 3.1.4). It identifies the white regions of the image and produces a contour map and hierarchy describing the white regions. Since the white regions of the images are the objects as identified by the above process, this results in a preliminary set of object locations. Next, bounding rectangles are drawn using the “cv2.boundingRect” function.

Figure 12 and Figure 13 show the results of this process for the spectral residual and fine-grained saliency methods, respectively, where the process successfully localized all but one target (targets included a child mannequin, an adult mannequin, three boxes, and a vehicle). Figure 12 shows the one target (i.e., the child mannequin) not localized and not found. Additionally, the algorithm localized other objects near the track, e.g., an orange cone. The algorithm successfully localized objects in the far distance, including a flag, a blue sign, and the white object located at the left side of the image. Additional analysis is necessary to discard objects that are clutter from OOIs.

Appendix A shows the use of this process to detect objects in NIR imagery at a range of 4,000 feet.

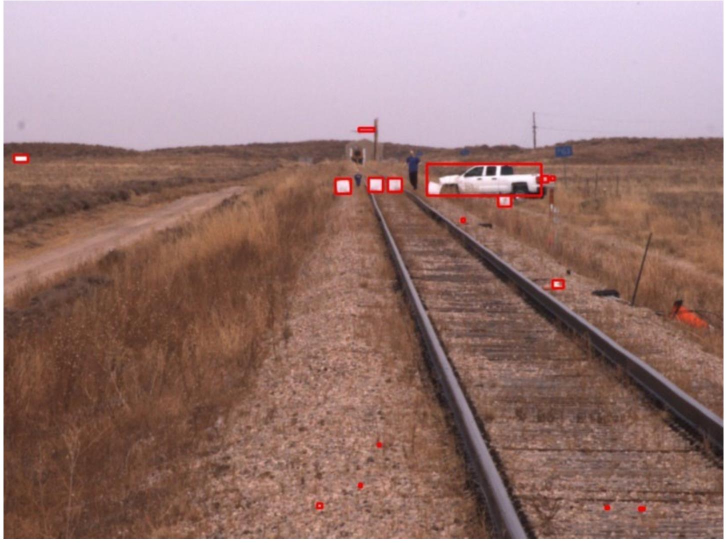


Figure 12. Spectral residual object localization

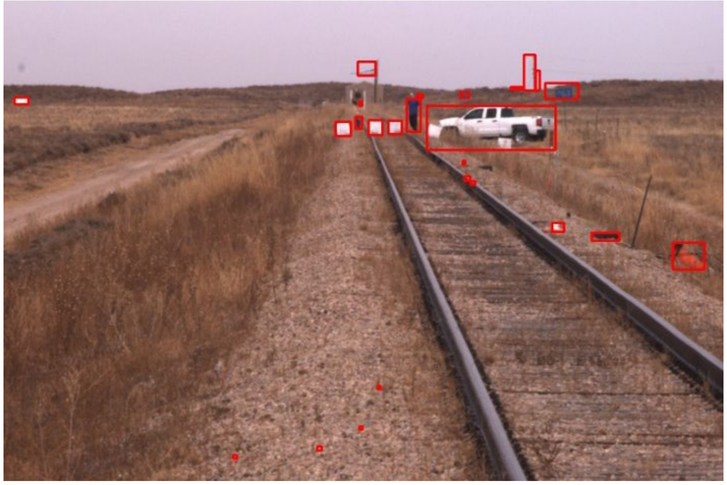


Figure 13. Fine-grained object localization

3.1.6 Railroad Track Segmentation

The team tested an additional algorithm (i.e., algorithm K) for railroad track segmentation. This algorithm combines an analysis of low-level details with a high-level context analysis to capture the larger image semantics (Yu, et al., 2021). Figure 14 shows an example output image.

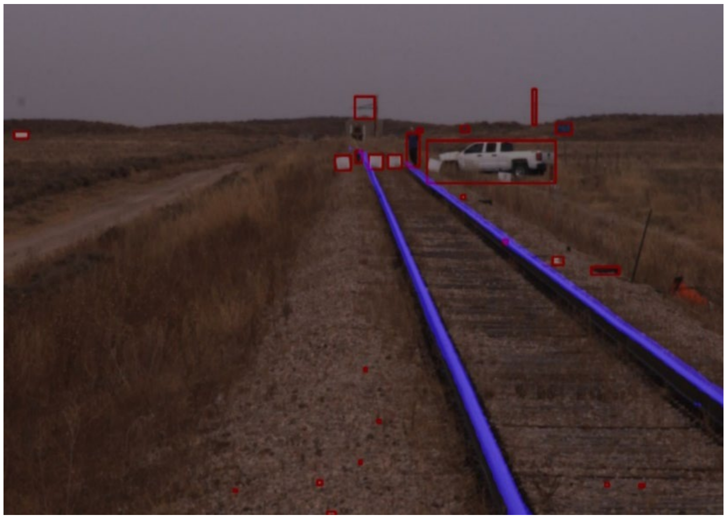


Figure 14. Object localization and rail segmentation output from input image shown in Figure 4

3.2 Autoencoder Neural Networks for Path Clear Detection

In the last 10 to 20 years, significant progress has been made in using deep neural networks (DNNs) for image classification, image segmentation (i.e., deciding which pixels belong to which objects), object detection, and anomaly detection. The research team considered the possibility of using DNNs for clear path detection.

The first step in using DNNs to detect obstructions in the foul volume is to locate the train tracks. This can be seen as either an image segmentation problem or an object detection problem. Once the track location is known, the second step is to detect whether the tracks are clear (a normal condition) or if there is an obstacle in the foul volume (an anomaly). If there is an obstacle in the foul volume, the distance to the obstacle must be determined. The team considered the development of an autoencoder-based track clear detector which, when given an image of the track, can determine if it is obstructed.

Autoencoders are a type of feed forward neural network and can be thought of as implementing a function (in the mathematical sense). Given an input image, an autoencoder will compress the input into a smaller dimensional space and then try to reconstruct the original image from the compressed representation. An autoencoder is a complex way of approximating the mathematical identity function (i.e., a function in which the output equals its input). The formula for the identity function is simply $f(x) = x$.

To explain the use of approximating a function as trivial as the identify function, it is necessary to explain how autoencoders work. Suppose an autoencoder takes as input a 100 pixel by 100 pixel grayscale image. Such an image can be thought of as a point in a 10,000-dimensional space (since $100^2 = 10,000$ and each grayscale pixel can be represented by a single number). An autoencoder does not simply copy its input to its output, as a copy and paste would. Instead, it first compresses the image into a smaller dimensional space called the latent space. The dimension of the smaller space is called the latent dimension, and it is essential that the latent dimension is smaller than the dimension of the input space.

An autoencoder is composed of two parts, an encoder and a decoder. The encoder compresses an input, X, into a representation, Y, in the latent space. The decoder takes an element Y of the latent space and produces an output X_1, which is an attempt at a reconstruction of the original input X.

Autoencoders do not work by a standard, general compression algorithm. They can only faithfully reproduce data similar to the training data, which is key to why they work as anomaly detectors. When training an autoencoder, numerous images of “normal" data are presented to it; in this case, images of train tracks where the foul volume is clear. The machine learning algorithm is forced to learn useful ways of representing salient features of the training data because the latent dimension is smaller than the input dimension.

Because an autoencoder can faithfully reproduce data similar to training data, and because it cannot do so for data unlike training data, a comparison of an input image X with the autoencoder output X_1 can determine if X is “normal” or not (i.e., whether or not the foul volume is free of obstacles). This is done by comparing how well X_1 matches X.

3.2.1 Benefits of Autoencoder for Path Clear Detection

There are at least two primary reasons why an autoencoder is an appealing approach for anomaly detection. First, the amount of data representing normal instances vastly outweighs the amount of data representing anomalies. Second, the number of different types of obstacles that could present a hazard to a train is unbounded. As discussed in Section 2.1.1, there is no possible way to specify in advance all the different possible obstacles that could show up. It would be enormously difficult, if not impossible, to train a classifier with enough categories to cover all possibilities.

3.2.2 Current Progress in Training an Autoencoder

As part of this project, researchers trained an autoencoder using railroad-provided video taken from the rear of a train. Several gigabytes of videos of clear tracks were provided by the railroads and the team extracted frames from those videos to use for training a deep neural network. As part of this effort, researchers trained an autoencoder on a subset of the data currently available; use of all available training data was not necessary for this research.

Each image used in training the DNN was a picture of a section of the track, cropped to 140 pixels by 140 pixels, and converted to grayscale. In Figure 15, Figure 16, and Figure 17, the training images are shown at the top and the autoencoder-reconstructed images are shown below. Figure 15 shows five examples of images before (top) and after (below) they were sent through the prototype autoencoder, including only straight track.

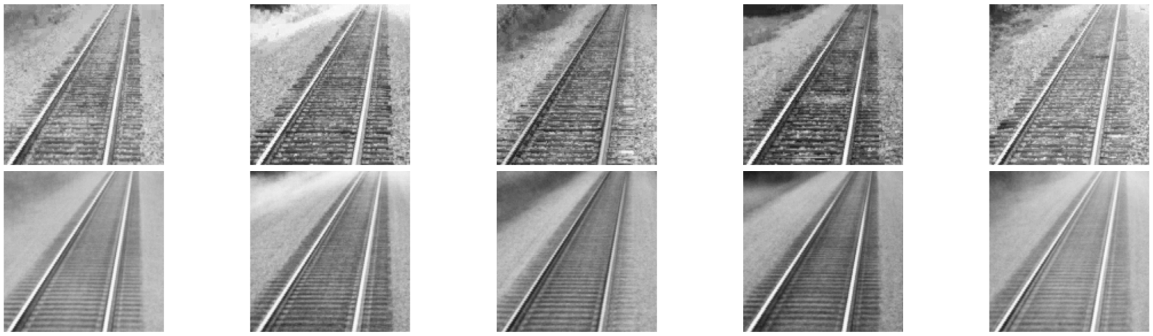


Figure 15. Prototype autoencoder results, straight track

Figure 16 shows five examples of images before (top) and after (below) they were sent through the prototype autoencoder, including both straight and curved track.

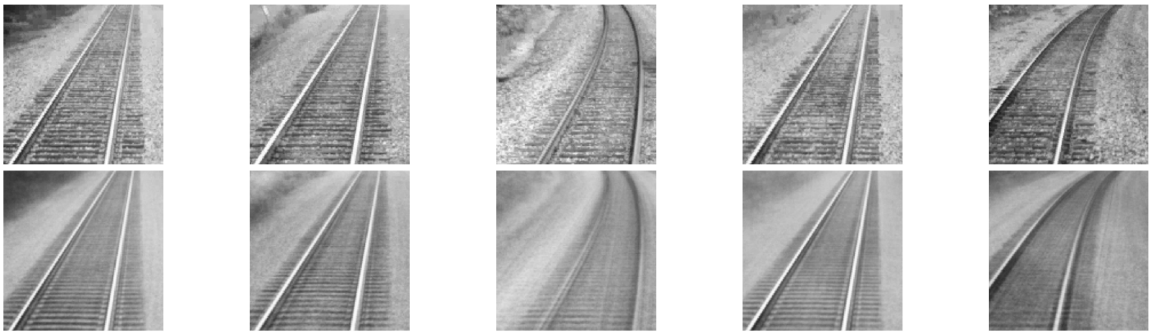


Figure 16. Prototype autoencoder results, curved and straight track

For curved tracks, the autoencoder produced a slightly more blurred image than for straight tracks. Curved tracks are not inherently more difficult for the machine learning algorithm to handle. Rather, the curved tracks appear blurred because only a few of the training images were of curved tracks. Once the model is trained with the rest of the available data, curved tracks should present no difficulty.

Figure 17 shows five more pairs of images, with a variety of shadows on the track.

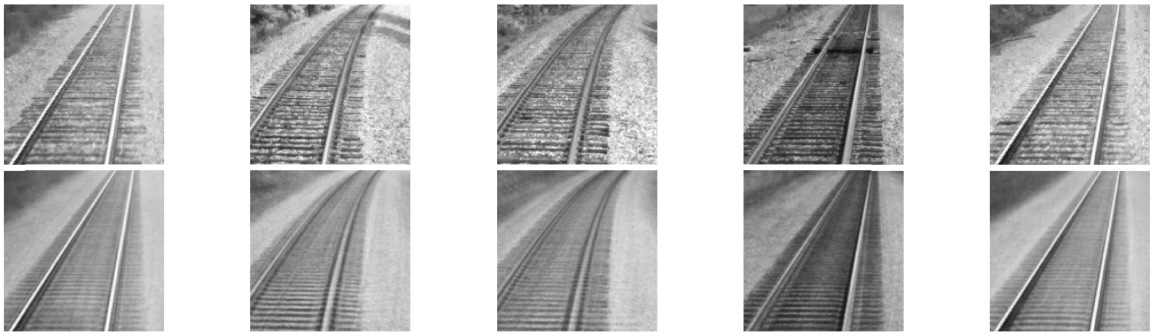


Figure 17. Prototype autoencoder results, with shadows

Shadows within images did not produce more blurred images, therefore the impact of shadows is not significant.

To test the ability of the autoencoder to detect obstacles, the team added obstacles artificially to images. Three types of obstacles were added (i.e., a fallen tree trunk, a car, and a person). The obstacles added artificially were not to scale. It is expected that the autoencoder will have more difficulty reconstructing small obstacles than large ones. This difficulty is addressed in Section 3.2.2.2 below.

Figure 18 shows five examples of images with obstacles artificially added before (top) and after (below) they were sent through the prototype autoencoder.

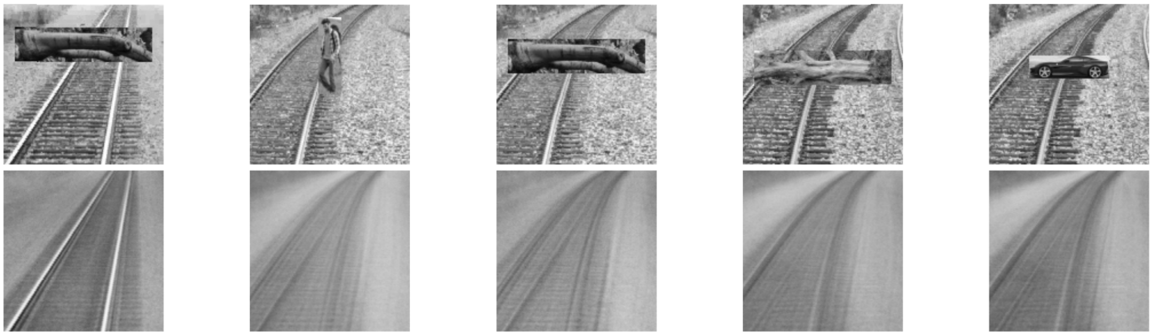


Figure 18. Autoencoder reconstructed images containing simulated obstacles, Test 1

Since an autoencoder can only faithfully reconstruct images similar to training objects, the autoencoder trained and studied as part of this effort wasn't able to reconstruct the simulated obstacles at all, suggesting an obstacle detection potential. Rather, as seen in Figure 18, it made the obstacles disappear entirely and attempted to reconstruct the train tracks based on what was visible.

Because the obstacles were not reproduced at all, the algorithm would indicate the presence of an obstacle. Additional tests of simulated obstructions are seen in Figure 19 and Figure 20.

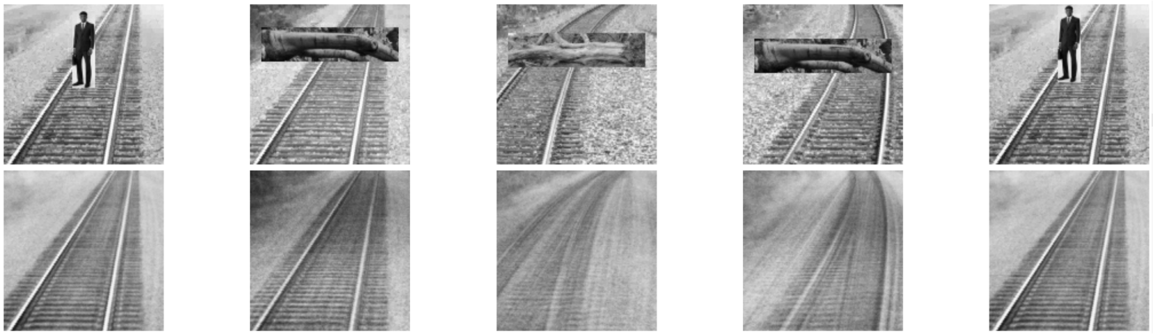


Figure 19. Autoencoder reconstructed images containing simulated obstacles, Test 2

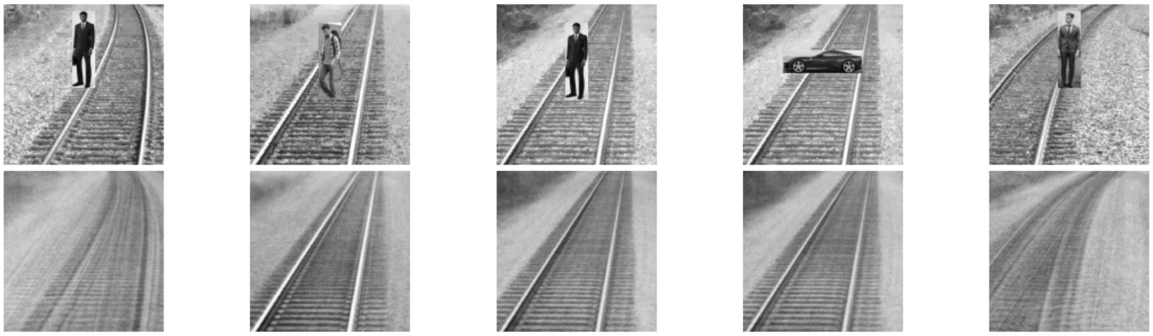


Figure 20. Autoencoder reconstructed images containing simulated obstacles, Test 3

From the results shown here, the team concluded that an autoencoder has obstacle detection potential.

3.2.2.1 Training with Images Containing Shadows

The team only completed preliminary work using images with shadows or switches. Researchers trained an autoencoder using the 1,000 images used in the first training as well as an additional 400 images containing more complex scenarios, including shadows. The results shown in Figure 21 show the performance of this autoencoder on images with simulated obstacles.

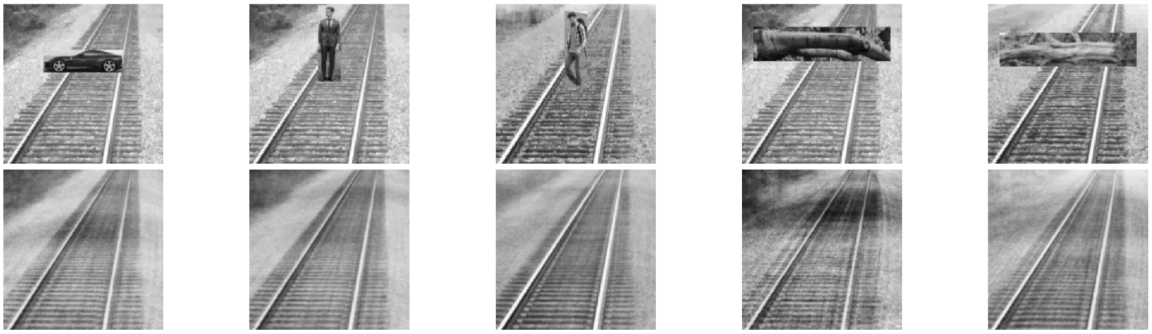


Figure 21. Autoencoder reconstructed images containing simulated obstacles with shadows

The addition of complex images introduced notable changes in the behavior of the autoencoder. The dark tree trunk in the image second from the right in Figure 21 results in a shadow-like effect in the autoencoder output. Similarly, the car in the first image on the left in Figure 21 results in a similar shadow-like effect. As shadows are the closest thing in the training dataset to the simulated obstacles, the team anticipated this behavior.

3.2.2.2 Difficulty with Apparently Small Objects

Given a working autoencoder, an anomaly detector is made by picking a threshold value, ρ. Images that the autoencoder reconstructs with a fidelity ρ or better will be declared as normal (i.e., obstacle-free). Images reconstructed with a fidelity worse than ρ will be considered to contain an obstacle. By adjusting the value of ρ, the false negative (i.e., indicating there is not an obstacle when there is) rate can be decreased. This causes a corresponding increase in the false positive rate (i.e., indicating there is an obstacle when there is not). Conversely, the false positive rate could be decreased, making the false negative rate increase.

The team did not determine the potential for the autoencoder to detect either small or distant obstacles as part of this effort. The farther an obstacle is from the train, the smaller of a difference the obstacle will cause in the autoencoder's output. Greater sensitivity to small differences in an image from its reconstruction will result in more false positives. The same problem is expected with small obstructions.

4. Classification Neural Networks

Although the basic question of whether the track ahead is clear is most important for the facilitation of ATO train movement, the use cases listed in Section 2.1 require several key types of objects to be explicitly classified as people, vehicles, or livestock. Since neural network-based classification processes are "blind" to classes of objects they are not explicitly trained to detect, the use of these types of networks in establishing high confidence that the train path is clear is limited to the set of object classes which the onboard system is trained to detect. This will be constrained by the capacity of onboard processing power. As the list of common objects for classification grows, so does the latency of the processing. A possible solution to mitigate the required onboard processing needs is to only attempt classification of known objects detected by the clear path detection algorithm.

Researchers evaluated several classification neural networks for use in SP classification tasks, and in this effort the sole focus was on the person class. The team evaluated networks on accuracy, precision, and recall (see Section 4.2), as well as network rates of false positive, false negative, true positive, and true negative. The team then combined several networks in a committee structure and further evaluated them to assess the potential for increased performance.

Classification neural networks require curated training and evaluation data sets. Every image or video in the data set must be labeled as to whether it does or does not contain each object on which the network is going to be trained. Depending on the specific neural network, the object may have to be in the image as well. Errors in the training data set can substantially reduce the reliability of the resulting analysis.

Researchers evaluated 10 different neural networks during this project representative of the state-of-the-art for neural network classification algorithms at the time this project was conducted.

4.1 Convolutional Neural Network Introduction

Artificial neural networks are complex, nonlinear computing systems. As the name implies, the functionality of neural networks is somewhat like the biological brain. Artificial brain cells within neural networks display emergent behavior through the interconnections between the artificial brain cells. The artificial brain cells can learn and classify data characteristics after a training process. Training the artificial brain cells is an iterative process of inputting known data into the neural net, observing the results, and adjusting the network parameters so the network converges on the correct results. As in the brain, neural networks are comprised of many neurons, referred to as nodes. Nodes are arranged in layers and are linked by connections. These connections are weighted by a scalar value which is iteratively adjusted in training to optimize the network results. Any given artificial neural network is comprised of three main parts: an input layer, a hidden layer, and the output layer. All data input into a neural network, such as image pixel values, is fed into the input layer. In the case of images, all pixels of the image are input in parallel, i.e., each pixel is simultaneously input into its own input node. The output layer is responsible for the final probabilistic prediction. Unlike the input and output layers, the hidden layer is comprised of not one but multiple layers of parallel nodes. Every node in a hidden layer is connected to every node in the hidden layers before and after it.

All neural networks evaluated under this project are of the convolutional neural network (CNN) type. CNNs are specifically tailored for image processing tasks. The key difference between a standard neural network and a CNN is in the operation performed by nodes at each layer. In a standard neural network, the operation at a node is a series of summations based on the previous layer's inputs and connection weights (this is called a fully connected neural network). In a convolutional neural network, the operation by nodes is a convolution of an input matrix (i.e., an image) with a smaller kernel of weights (i.e., a filter). A single layer of a convolutional network is comprised of multiple parts, generally consisting of a convolutional block, pooling layer, and rectified linear unit (ReLU) activation.

The kernel, or filter, of a convolutional block convolves across the input image to create a smaller resulting image called a feature map. The elements of the filter matrices are called the filter weights. The filter weights are iteratively adjusted in network training to give an optimal response at each layer. Figure 22 illustrates the convolution process. The input is shown on the left of the figure, with the 3 by 3 convolution filter overlayed in green. The result, shown on the right of the figure, is the sum of the input and filter element products.

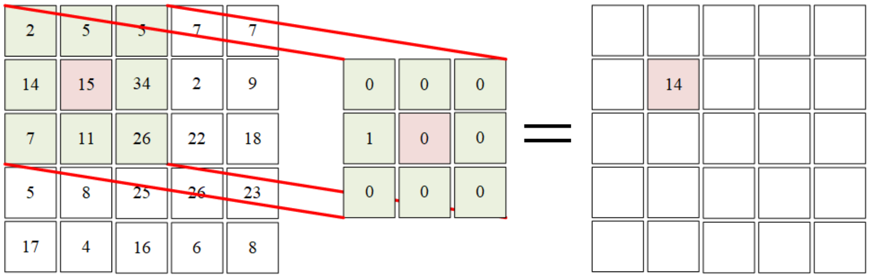


Figure 22. Example convolution of an input matrix and a 3 by 3 filter

A given convolutional layer generates a series of output feature maps that are generally too large for practical purposes, so they are further downsampled by a pooling function. The goal is to substantially downsample data at each network layer to arrive at a classification in a practical number of layers. Downsampling also reduces the number of parameters, which reduces the overall computational cost. The most widely used form of pooling is max pooling. It operates by applying a 2 by 2 filter to the feature maps. The 2 by 2 filter moves by two pixels as it scans across the maps, and it outputs the maximal value in the four-pixel neighborhood. The result is a series of feature maps that have been downsampled by 75 percent. Figure 23 illustrates the pooling process. The maximal pixel value from each 2 by 2 colored region is passed to the smaller resulting matrix.

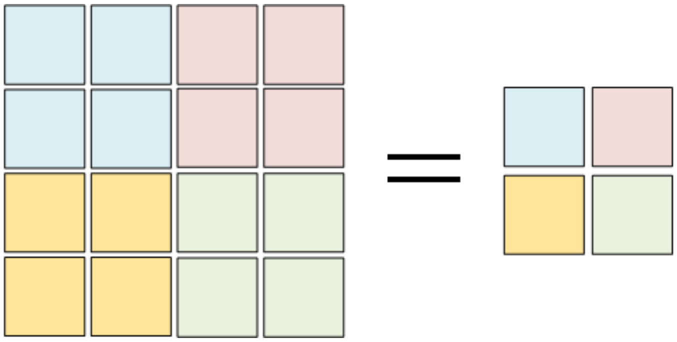


Figure 23. Example of the pooling process resulting in a 75 percent reduction of data

After the pooling process is complete, a ReLU activation function is applied. Neural networks, whether fully connected or convolutional, rely on activation functions to make the networks nonlinear. The ReLU function is simple; after pooling, each element of a feature map is cleared to zero if the value of the element is negative. The use of ReLU activation further reduces data in the feature maps, making the network lighter in computation.

After several repetitions of the convolution, pooling, and ReLU cycle, data is downsampled enough that it may be presented to the output layer. The last layer of feature maps is transformed into a one-dimensional feature vector that is presented to the output layer for final classification. The output of this final classification network is a vector with the number of elements equaling the number of classifications, and which contains a distribution of real values. A transformation function is applied to the final output vector to transform the values to a range between zero and one, where all the entries add up to one. This gives a more intuitive statistical representation of the result in a probability distribution.

4.2 Training Classifier Neural Networks

The 10 different neural networks evaluated during this project supported the classification of many discrete object classes. These classes show a high level of detail, containing classes such as tie, baseball hat, and glove. Although this level of detail is impressive, it often creates undesirable results for the identification of a single object class, such as a person. For this reason, the evaluated networks were trained using a training database containing images of only two classes: “person” and “non-person.” The people in these images were shown in various orientations and environments. Each image labeled “person” contained only one individual. The images containing no people were of various objects (e.g., motorcycles, cars, airplanes) in different environments. The training image dataset was assembled from annotated images contained in the Massachusetts Institute of Technology image database LabelMe, and Microsoft’s Common Objects in Context (COCO) image database. The final training database for this project contained a total of 406 annotated images, half containing persons and half containing non-persons.

4.3 Classifier Neural Networks Evaluation Approach

The team tested the neural networks against an evaluation dataset of 266 images assembled from the LabelMe and COCO datasets. This combined dataset contained 133 images containing persons and 133 containing no persons. The training and evaluation image datasets were independent and did not share any images. The networks were evaluated on their accuracy, precision, and recall performance against this evaluation dataset. Network rates of false positive, false negative, true positive, and true negative were also evaluated against the dataset. Each neural network outputs the probability of an image belonging to either the “person” or “non-person” class in the form of two probabilities (since there are two possible classes) whose sum is equal to 1. The images receive the label of the highest probability prediction.

When evaluating a neural network, its precision is measured by the percentage objects correctly labeled as a detection vs the total number of objects labeled as a detection. This is calculated as the number of true positive detections divided by the total number of detections:

$$precision = \frac{true\ positive}{true\ positive + false\ positive}$$

The recall of a neural network is the percentage of correct positive detections out of the total number of items that should have been detected. The recall is calculated by dividing the number of objects correctly labeled positive by the number of objects that should have been labeled positive:

$$recall = \frac{true\ positive}{true\ positive + false\ negative}$$

The accuracy of a neural network is the percentage of the positive detections that are correct. The accuracy is calculated by dividing the number of objects correctly labeled by the number of all labeled objects:

$$accuracy = \frac{true\ positive + true\ negative}{true\ positive + true\ negative + false\ positive + false\ negative}$$

The precision and recall of a particular classification neural network are engineering tradeoffs. As the precision of the network is increased, the recall generally decreases. The opposite is also true; as recall is increased, precision of the network decreases.

4.4 Neural Network Evaluation Results

This section contains a description of the performance of each classification neural network tested. The confusion charts shown in this section illustrate the number of true positives, true negatives, false positives, and false negatives, categorized as follows:

- True positive: the model correctly identifies a person in an evaluation image
- True negative: the model correctly identifies the lack of a person in an evaluation image
- False negative: the model incorrectly identifies an evaluation image as lacking a person
- False positive: the model incorrectly identifies an evaluation as containing a person when it does not

In the confusion charts, true positives and true negatives are represented by blue shades where a darker blue indicates a larger number of correct detections. Red represents the false positives and false negatives, where a darker red indicates a larger number of incorrect detections. In a well performing model, there are large numbers of true positives and true negatives and only a small number of false positives and false negatives. Minimizing the number of false negatives is of the most interest in this effort.

Ten neural networks were evaluated for their object detection capability, and were labeled Network A to Network J. The neural networks were anonymized as the purpose of this study is a preliminary evaluation of the feasibility of constructing an SP and this evaluation should not be used in comparing the neural networks for other purposes.

4.4.1 Network A

The structure of Network A contains 25 convolutional layers. It was the first CNN to support the use of GPUs to boost performance. Due to its simple structure, Network A requires relatively low computational power, but also showed the lowest accuracy, precision, and recall scores of any neural network tested during this project. Figure 24 below shows the confusion chart of Network A. Of the 133 images containing persons, Network A incorrectly classified 59. Therefore, the false negative rate was 44 percent.

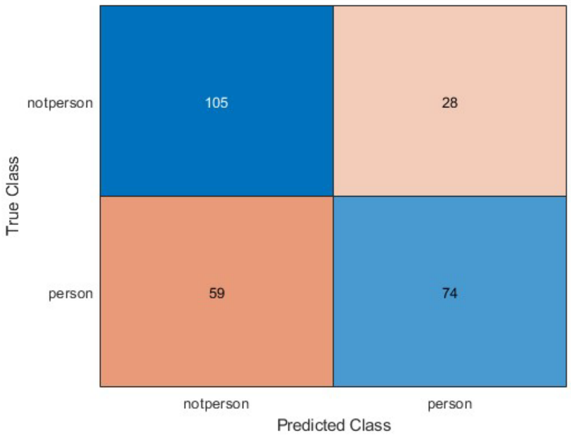


Figure 24. Confusion chart of Network A

4.4.2 Network B

Network B is the CNN backbone of popular object detection methods. It differentiates itself from earlier, similar networks with the use of residual connections and the addition of more layers, containing 53 layers in total. Residual connections skip the immediately following convolutional layers and connect directly to more distant layers. In testing during this project, Network B performed well, scoring near the top on precision and in the upper middle on accuracy and recall. Figure 25 below shows the confusion chart of Network B. The false negative rate of this network was 35 percent.

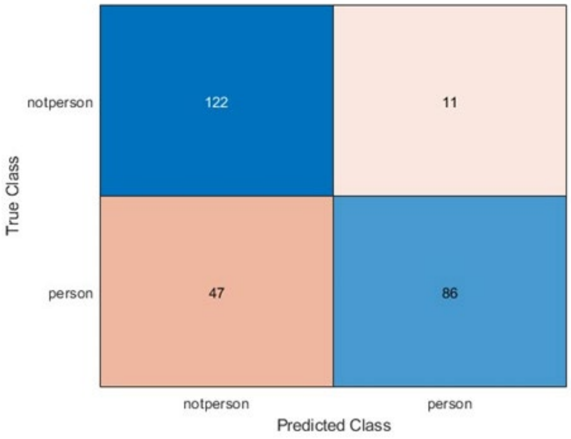


Figure 25. Confusion chart of Network B

4.4.3 Network C

Network C was designed to maintain acceptable performance, both in terms of computational power required and in terms of accuracy, when the depth, width, and resolution are scaled up. For this testing, only the baseline model of Network C was available, though versions of this model with more layers exist. Network C saw identical results to Network B, as shown by the confusion chart in Figure 26 below. The structures of many neural networks are quite similar, in fact many networks are built on the foundations of others; this is a likely explanation for the observed results. The false negative rate of this network was 35 percent.

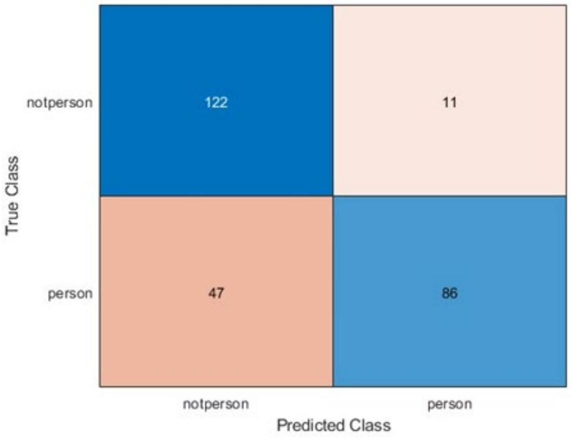


Figure 26. Confusion chart of Network C

4.4.4 Network D

Network D is a CNN that is based on a similar architecture to Network E. Network D uses several changes to the typical convolution and pooling techniques to create a more efficient and accurate network architecture. Network D uses convolution filters which downsample data at a more aggressive rate than similar networks, reducing overall computational cost. This network also leverages average pooling filters, instead of the often-used max pooling. In this testing, Network D was an outstanding performer. It earned the best accuracy and precision scores and performed well in recall. As a top performer, Network D was chosen as the primary network for the fusion committee described in Section 4.6. Figure 27 below shows the confusion chart of Network D. The false negative rate of this network was 16 percent.

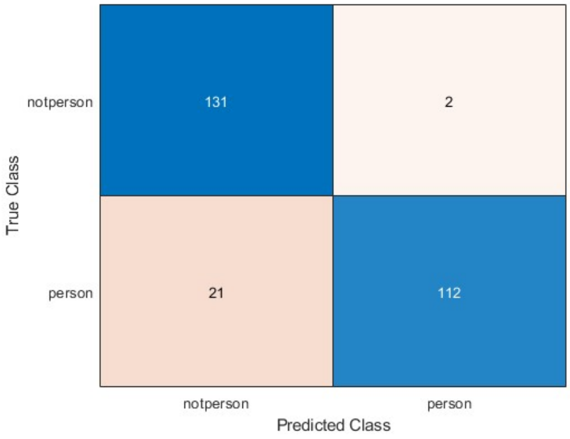


Figure 27. Confusion chart of Network D

4.4.5 Network E

Network E can be considered a hybrid between Networks D and B. Network E leverages many of the convolution and pooling strategies of Network D, while also using residual connections seen in Network B. This testing showed this model to be effective, with results for the tests in the upper middle of the group tested. However, both Network G and Network H outperformed this model, as did Network D. Figure 28 below shows the confusion chart of Network E. The false negative rate of this network was 23 percent.

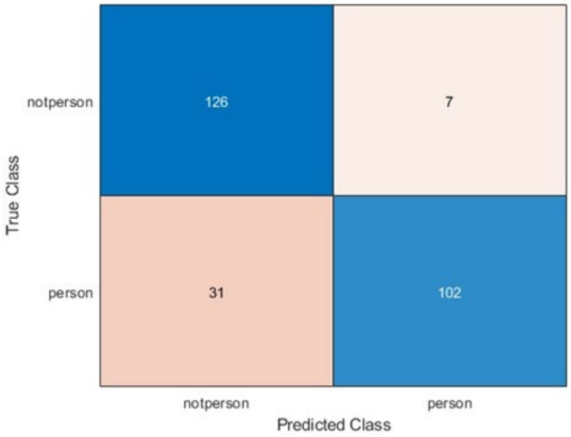


Figure 28. Confusion chart of Network E

4.4.6 Network F

Network F shows novelty over other networks in its implementation of the convolution process. Most networks perform only one convolution type per convolution layer. Network F performs three separate convolutions in sequence per convolution layer. The convolutions can either decrease or increase the number of resulting parameters proportional to an expansion factor. This factor was one of the parameters adjusted in training to achieve optimal results. Network F performed well in this testing with acceptable accuracy, precision, and recall results. Figure 29 below shows the confusion chart of Network F. The false negative rate of this network was 20 percent.

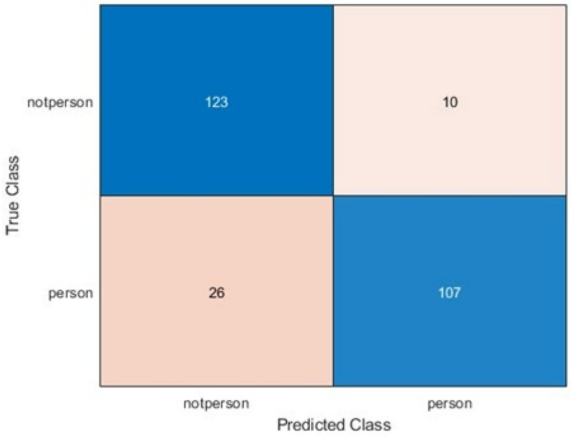


Figure 29. Confusion Chart of Network F

4.4.7 Network G

Network G's model was created to solve what is called the "vanishing gradient" problem. Before the release of Network G, CNNs had an issue of strongly diminishing performance returns for each additional network layer past a certain layer number. Network G solves this problem by using a new network layer called the residual block. This process is similar to the residual connections used in Network B. Residual connections skip several convolutional layers at a time. The layers skipped by a connection are considered the residual block. Residual blocks can be selectively skipped if that layer is a hindrance to the overall performance which helps it gain increasing accuracy with additional network layers. Network G performed well in testing, with accuracy and precision scores in the 90s and a recall in the 80s. The only network that bested it in recall is the more powerful Network H, which is a larger version of Network G. Figure 30 below shows the confusion chart of Network G. The false negative rate of this network was 12 percent.

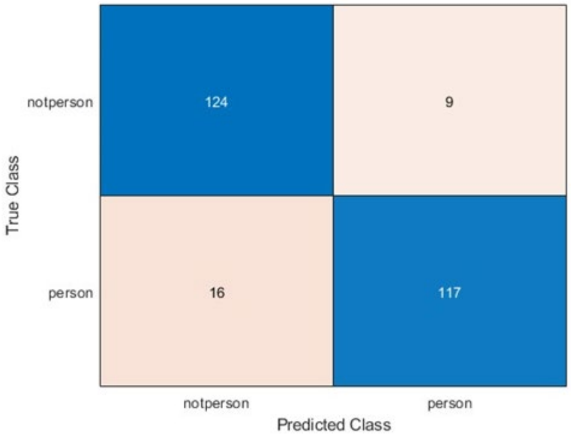


Figure 30. Confusion Chart of Network G

4.4.8 Network H

Network H is a larger version of the neural network seen in Network G. It is one of the best performers in this testing, seeing good accuracy, precision, and best overall recall of networks tested. Figure 31 below shows the confusion chart of Network H. The false negative rate of this network was 14 percent.

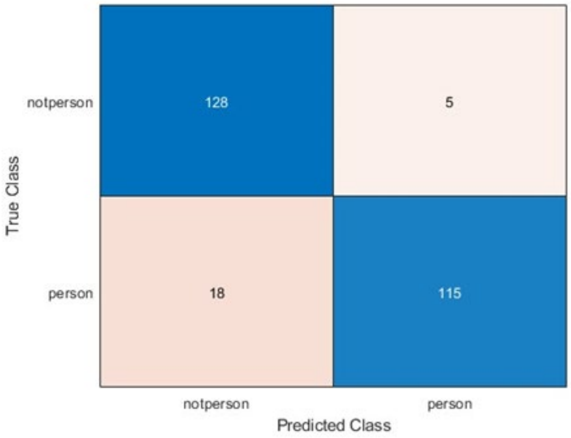


Figure 31. Confusion Chart of Network H

4.4.9 Network I

Network I was designed to reduce the required network bandwidth when training the network using distributed training (i.e., training over a network). This was done by generally minimizing the size of the network architecture. Network I primarily accomplishes this by aggressively reducing parameters in each convolutional layer, leading to a rapid downsampling of the input data size. In evaluation, Network I showed better performance than Network A in all categories; however, Network A is the only network it outperformed. In recall, Network I resulted in a middle of the pack score. Figure 32 below shows the confusion chart of Network I. The false negative rate of this network was 29 percent.

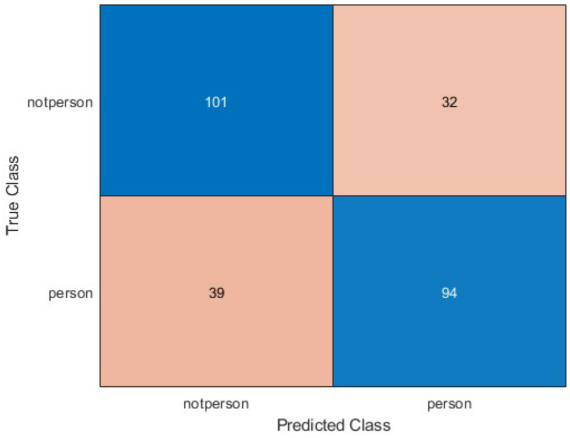


Figure 32. Confusion Chart of Network I

4.4.10 Network J

Network J was directly based on Network A and was designed to be an improvement over the original. The structure of the network greatly resembles Network A with a single, linear architecture. The main difference between the two networks lies in the convolution layer and convolution filter size. Network J uses substantially smaller convolutional filter kernels compared to Network A, making Network J more discriminative in its results. Network J substantially improved Network A's accuracy and precision with scores increasing from 67 to 76 and 73 to 91, respectively. On recall, there was only a slight improvement from 56 to 58. While performance increased, Network J took roughly 16 times longer to train than Network A. Figure 33 below shows the confusion chart of Network J. The false negative rate of this network was 42 percent.

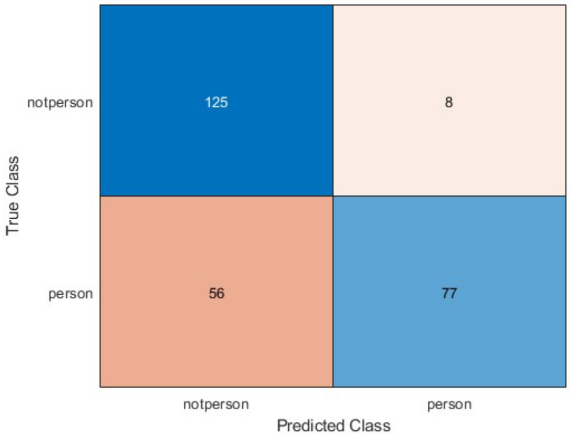


Figure 33. Confusion Chart of Network J

4.5 Summary of Neural Network Person Detection Performance

Table 1 shows the name, accuracy, precision, and recall performance of all networks tested. Network D, Network G, and Network H were top performers.

Table 1. Performance of Neural Networks

Network	Accuracy (%)	Precision (%)	Recall (%)
Network A	67	73	56
Network B	78	89	65
Network C	79	89	65
Network D	91	98	84
Network E	86	94	77
Network F	86	91	80
Network G	91	93	88
Network H	91	96	86
Network I	73	75	71
Network J	76	91	58

Table 2 below shows the individual false positive and false negative rates of each network. These two error types have the most direct impact on rail network performance and safety. High false positive error rates (incorrectly classifying an object as a person) will cause unnecessary train response and slow down train operations. High false negative rates (failing to classify a person) can result in safety risks since the appropriate train response may not be triggered. These results suggest that individual classification networks may not meet railroad safety standards. For all networks, the false negative rate is higher than the false positive rate. Future efforts should focus on minimizing false negative rates of individual networks.

Table 2. False Positive and False Negative Rates of Networks

Network	False Positive Rate (%)	False Negative Rate (%)
Network A	21	44
Network B	8	35
Network C	8	35
Network D	2	16
Network E	5	23
Network F	8	20
Network G	7	12
Network H	4	14
Network I	24	29
Network J	6	42

4.6 Neural Net Committee Algorithm

Researchers explored a neural network output fusion technique to improve classification performance. This output fusion technique is referred to as a committee algorithm. The neural network committee is comprised of multiple independent neural networks whose individual results are considered a vote in the collective task outcome.

4.6.1 Committee Network Selection

The primary emphasis in the committee network selection is finding algorithms that complement each other's performance. In other words, if a network fails to classify persons in a particular set of images, the network should be paired with one that does correctly classify persons within that same image set. Finding such complementary networks reduces the chance that the committee will be affected by the same source of error. Finding good complementary networks increases the efficiency of the committee. Combining networks with similar performances on a set of images is redundant since their output will be very similar. Removal of redundant networks within the committee by selecting only complementary networks reduces the computational cost of the committee and reduces computing hardware necessary on board an ATO train.

The team evaluated two methods of selecting complementary networks. In the first option for selection, the team identified the overall best neural net as the primary (Network D). The images which caused Network D to perform poorly (false positives and false negatives) were then presented to all other networks. The top two networks with the best performance on the set of Network D false positive and false negative images were selected as its complements (Network I and Network G) and added to the three-network committee. Figure 34 illustrates this first option committee selection process.

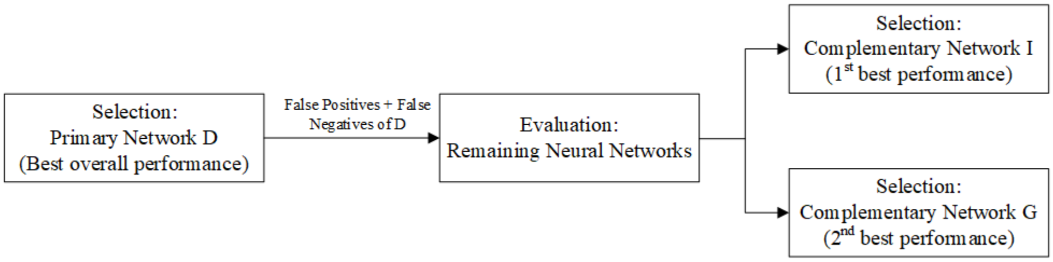


Figure 34. First option process for committee selection

In the second option for selection, the team identified a network as the primary (Network D) and then presented the images that caused Network D to perform poorly to all other networks. The first complement network was selected based on its performance on Network D's false positive and false negative images. This network is called the first complement (Network I). The images that caused Network I to perform poorly were then presented to the remaining networks. The second complement was found by evaluating all remaining networks against the false positive and false negative image database of Network I. The top performer was selected as the second complement (Network H). Figure 35 shows the second option committee selection process.

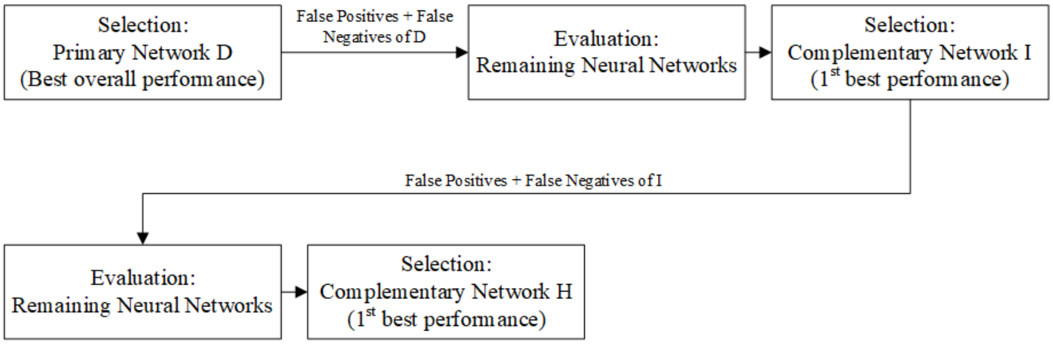


Figure 35. Second option process for committee selection

Table 3 below shows the network performance on the false positive and false negative images of Network D.

Table 3. Network Performance on Network D False Positive and +False Negative Images

Name of Network	Accuracy (%)	Precision (%)	Recall (%)
Network A	22	80	19
Network B	30	86	29
Network C	30	86	29
Network E	30	86	29
Network F	26	75	29
Network G	48	91	48
Network H	43	83	48
Network I	52	86	57
Network J	17	75	14

Network I was the overall best complement to Network D with the best accuracy and recall scores. Network G was the second best complement with the best precision score. Following the first option committee selection method, Network D, Network I, and Network G were selected for the committee.

Table 4 below shows the network performance on the false positive and false negative images of Network I.

Table 4. Network Performance on Network I False Positive and +False Negative Images

Name of Network	Accuracy (%)	Precision (%)	Recall (%)
Network A	48	53	46
Network B	63	74	51
Network C	73	78	72
Network E	76	84	69
Network F	77	81	77
Network G	82	83	85
Network H	85	89	82
Network J	69	84	84

Network H was the overall best complement to Network I, as it had the best accuracy and precision scores with a recall score that was close to the top. Following the second option committee selection method, Network D, Network I, and Network H were selected for the committee.

4.6.2 Committee Network Performance Evaluation

Researchers evaluated both the first and second option committee algorithms against the same 266 image evaluation dataset as the individual networks. The team evaluated two voting schemes of the committee algorithm output fusion, the majority vote and minority vote.

4.6.2.1 Majority Vote Committee Approach

The majority vote approach requires consensus from two of the three neural networks before an object is positively classified (e.g., if only one of three networks classifies an object as a person, the object is considered not a person since the majority of networks did not agree on a result). Figure 36 illustrates this process.

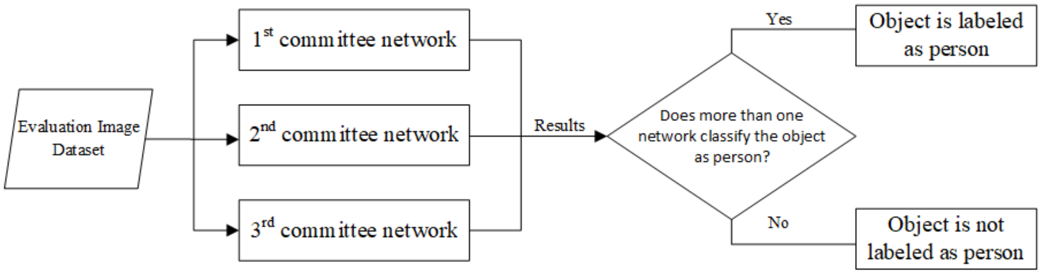

Evaluation Image Dataset
1st committee network
2nd committee network
3rd committee network
Results
Does more than one network classify the object as person?
Yes
Object is labeled as person
No
Object is not labeled as person

Figure 36. Illustration of the majority vote process

Table 5 shows scores of the majority vote committees and the independent Network D score. For the majority vote scheme in the Network D, Network I, and Network G committee, accuracy remained the same, while recall increased and precision decreased compared to Network D alone. In the Network D, Network I, and Network H committee, accuracy and recall increased and precision decreased slightly compared to Network D alone.

Table 5. Committee Results for Majority Vote

Neural Networks	Accuracy (%)	Precision (%)	Recall (%)
Network D	91	98	84
First option committee (Network D, Network I, Network G)	91	93	90
Second option committee (Network D, Network I, Network H)	92	96	89

Table 6 shows the false positive and false negative rates of the majority vote committees. Both committees saw a notable decrease in the false negative rate.

Table 6. False Positive and False Negative Rates of Majority Vote Committees

Neural Networks	False Positive Rate (%)	False Negative Rate (%)
Network D	2	16
First option committee (Network D, Network I, Network G)	7	10
Second option committee (Network D, Network I, Network H)	4	11

Figure 37 and Figure 38 below show the confusion plots of the two majority vote committees.

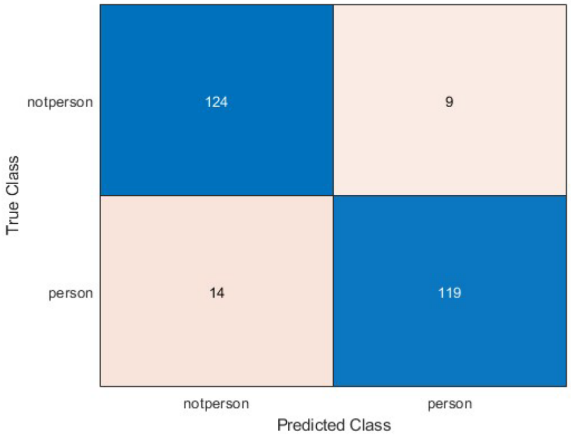


Figure 37. Confusion plot first option committee majority vote

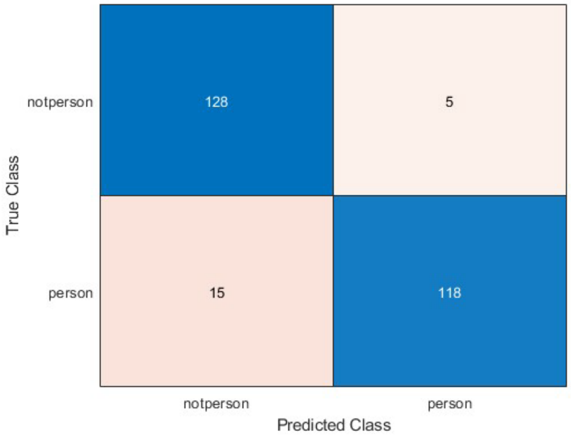
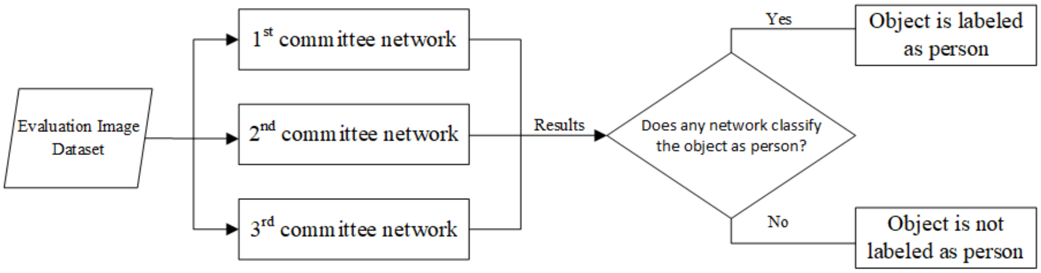


Figure 38. Confusion plot second option committee majority vote

4.6.2.2 Minority Vote Committee Approach

The minority vote approach requires a positive result from just one of the three neural networks before an object is positively classified (e.g., if just one of three networks classifies an object as a person, the object is identified as a person, since only one network vote is necessary). Figure 39 shows this process.



Evaluation Image Dataset
1st committee network
2nd committee network
3rd committee network
Results
Does any network classify the object as person?
Yes
Object is labeled as person
No
Object is not labeled as person

Figure 39. Illustration of the minority vote process

Table 7 shows the minority vote committee scores and independent Network D score. In the minority vote scheme, the two committees scored lower in accuracy and precision when compared to Network D alone, but both committees saw a more than 10 percent increase in recall score. Although Network D was the single best overall performer, the network saw a relatively low recall compared to its accuracy and precision. The committee output fusion method mitigated this weakness.

Table 7. Committee Results for Minority Vote

Neural Networks	Accuracy (%)	Precision (%)	Recall (%)
Network D	91	98	84
First option committee (Network D, Network I, Network G)	82	75	96
Second option committee (Network D, Network I, Network H)	82	75	95

Table 8 below shows the false positive and false negative rates of the minority vote committees. With the minority vote scheme, the first option committee saw a 12 percent decrease in false negative errors. This shows that committee type output fusion algorithms could be leveraged to increase safety of ATO operations. Future data analysis efforts should focus on refining the output fusion process. Larger committee size and different voting schemes may result in the further decrease of false negative rates.

Table 8. False Positive and False Negative Rates of Minority Vote Committees

Neural Networks	False Positive Rate (%)	False Negative Rate (%)
Network D	2	16
First option committee (Network D, Network I, Network G)	32	4
Second option committee (Network D, Network I, Network H)	32	5

Figure 40 and Figure 41 below show the confusion plots of the two minority vote committees.

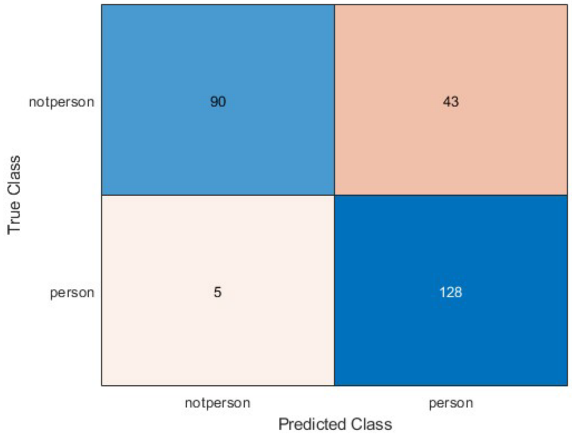


Figure 40. Confusion plot first option committee minority vote

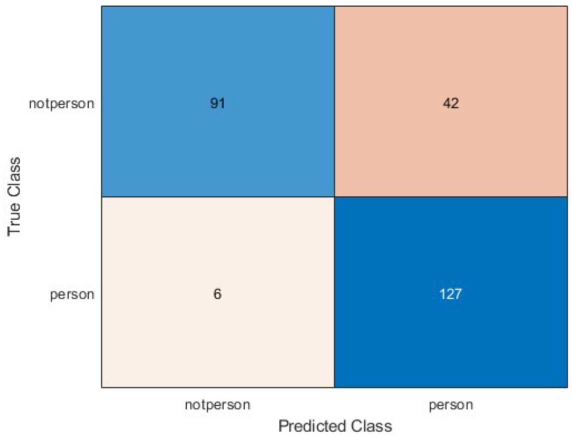


Figure 41. Confusion plot second option committee minority vote

5. Conclusion and Recommendations

In the ATO SP Data Analysis RP project, researchers evaluated the feasibility of using COTS software tools and algorithms in the creation of SP data analysis processes capable of supporting railroad automation use cases. The team investigated the priority processes of clear path detection and classification of people. Researchers analyzed data from a prior ATO SP RP project as part of the clear path detection effort.

The team investigated clear path detection as a possible solution to the difficulty of identifying every possible hazard that could occur in the rail environment. Researchers studied visual saliency-based and autoencoder-based clear path detection approaches for potential use in clear path detection. The preliminary results from both methods suggest that clear path detection is feasible for use as a component of an SP by providing class-independent hazard detection. In addition, the visual saliency approach produces a list of possible objects that could be fed to an object classification algorithm to further refine information related to objects.

Researchers evaluated 10 convolutional neural networks for use in the classification of people. Key findings from the evaluation include:

- The classification performance of any single network may be insufficient to meet confidence levels for operational safety.
- All classification networks evaluated showed a bias away from false positive results and toward false negative results. The average false positive rate of all networks was 9.3 percent, while the average false negative rate was 27 percent.

The output of the several neural networks was fused to explore potential classification performance benefits. The team evaluated two different committee algorithms that used two different voting approaches. Key findings of the committee algorithm evaluation include:

- The three-network committee algorithm approach saw a maximal reduction in false negative rates of 12 percent when compared to the performance of a single network.
- Output fusion techniques can potentially be leveraged to help meet operational safety levels.

Further refinement of both clear path detection and object detection is needed. Recommended future work includes:

- Investigation into reducing false negative error rates of classification processes – This includes determining the cause of high false negative rates compared to false positive rates observed in neural networks
- Implementation and evaluation of additional clear path detection and object detection approaches
- Expansion of and evaluation of classification neural networks to include the set objects and conditions of interest identified in the SP requirement documentation (Federal Railroad Administration, 2020)
- Collection of additional data in the railroad environment, including all environmental conditions in which an ATO train may operate – This work should build toward a

thorough sampling of the environment encountered across all the North American Class I railroads

- Creation of a properly annotated training and evaluation data set – This data set should be taken from the railroad environment (per the above recommendation) and carefully annotated for use in training and evaluating machine learning algorithms
- Evaluation of the neural networks using larger railroad-specific data sets
- Additional work on committee-based approaches to improving object detection reliability

In summary, the data analysis conducted in this effort suggests that an SP capable of benefiting railroad operations is possible, but additional work remains to fully prove the concept.

6. References

About OpenCV (2023). OpenCV.

Carsten Rother, V. K. (2004). GrabCut: Interactive Foreground Extraction Using Iterated Graph Cuts. *ACM Transactions on Graphics, 23*(3), 309-314.

Federal Railroad Administration (2020). *Automated Train Operations (ATO) Safety and Sensor Development* (Report No. RR 20-21). US Department of Transportation.

Hou, X., & Zhang, L. (2007). Saliency Detection: A Spectral Residual Approach. *2007 IEEE Conference on Computer Vision and Pattern Recognition.* Minneapolis.

Itti, L., & Coch, C. (2001). Computational Modelling of Visual Attention. *Nature Reviews: Neuroscience*, *2*, 194-203.

Montabone, S., & Soto, A. (2010). Human detection using a mobile platform and novel features derived from a visual saliency mechanism. *Image and Vision Computing, 28*(3), 391-402.

National Transportation Safety Board (2022). *Aviation Accident Preliminary Report, WPR22LA076*. National Transportation Safety Board, Washington, DC.

Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics, 9*(1), 62-66.

Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, Faster, Stronger. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* (7263-7271).

Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767.*

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-time Object Detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (779-788).

Wang, B., & Dudek, P. (2014). A Fast Self-tuning Background Subtraction Algorithm. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW),* (23-28).

Wang, Y., Want, L., Hen Hu, Y., & Qiu, J. (2019). RailNet: A Segmentation Network for Railroad Detectionv. *IEEE Access, 7*, (143772-143779).

Appendix A. Object Localization Example Images

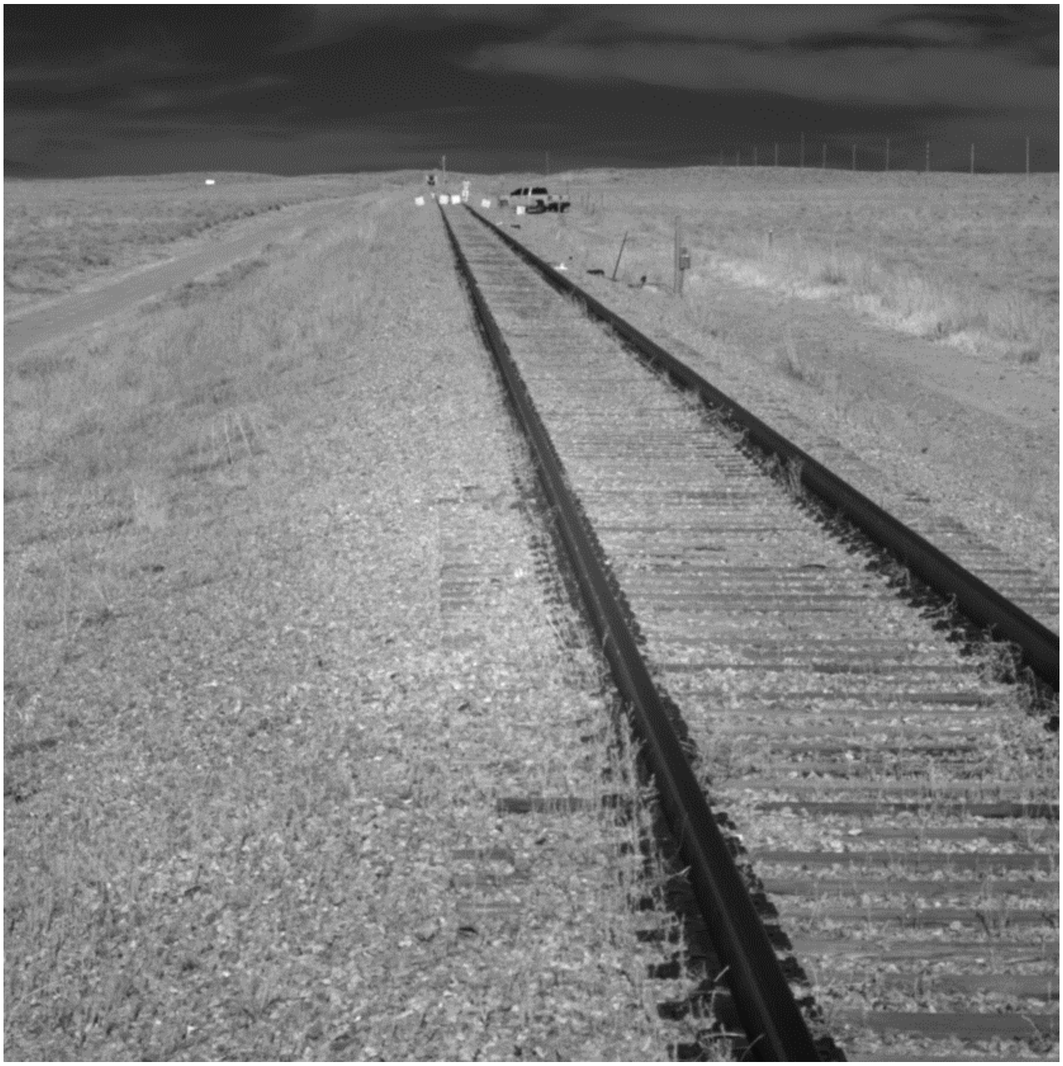


Figure A1. Sample IR input image

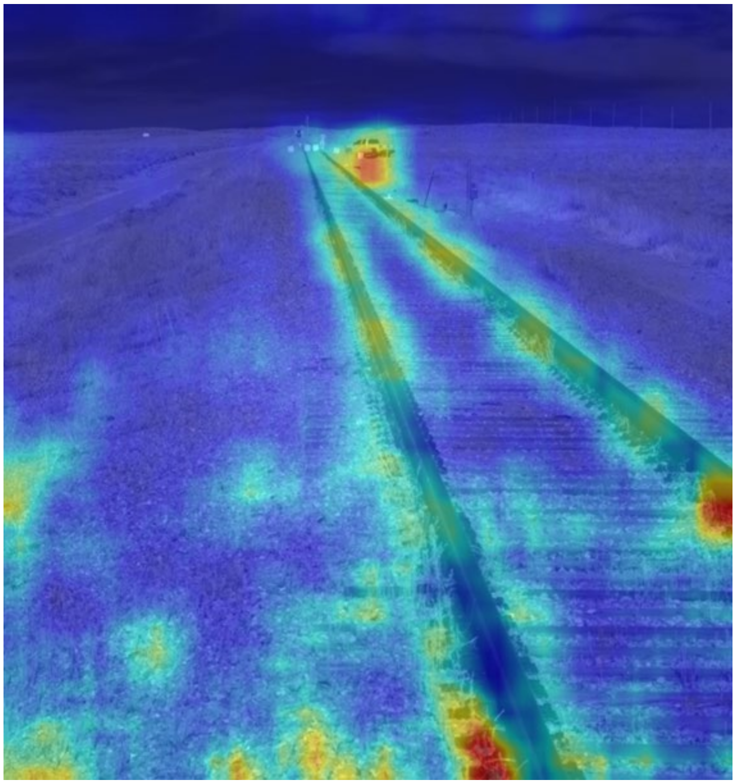


Figure A2. Spectral residual saliency map overlaid onto the input image

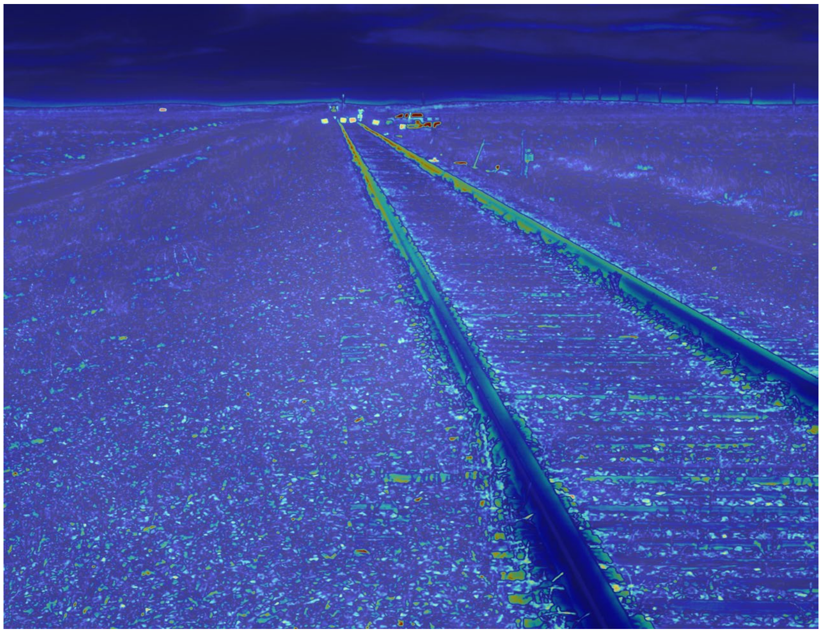


Figure A3. Fine grained saliency map for input image in Figure 5

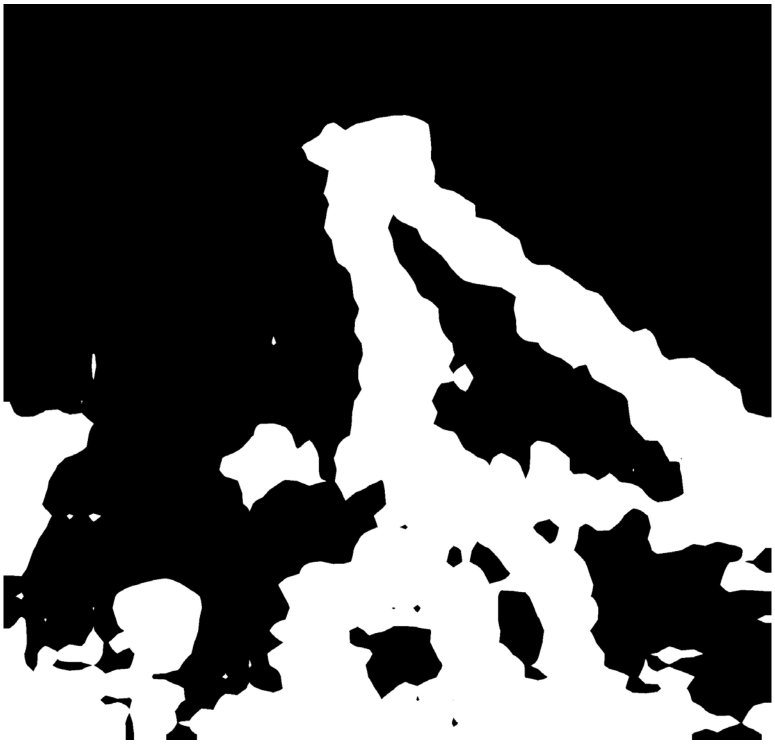


Figure A4. Spectral residual saliency map for input image

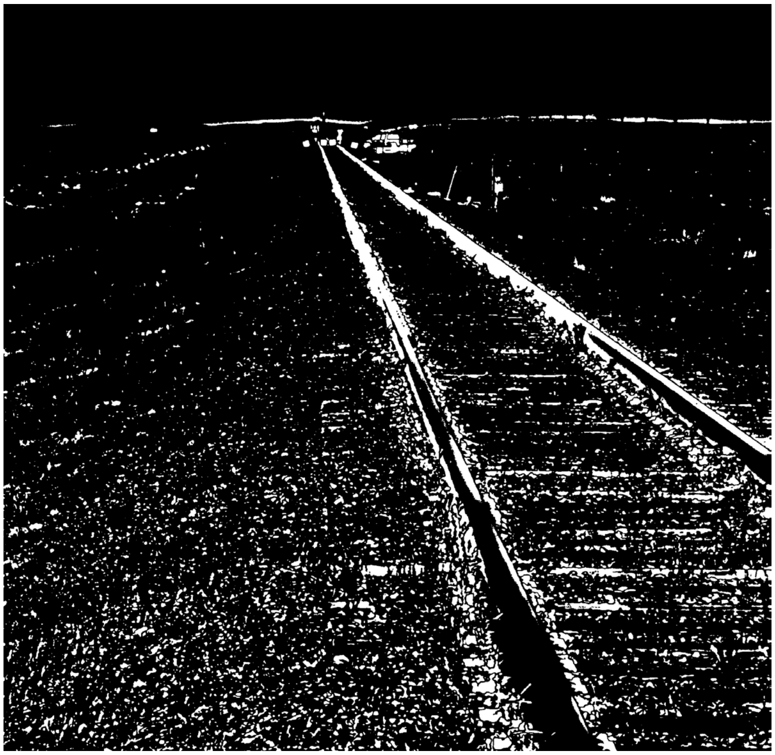


Figure A5. Fine grained saliency map for input image

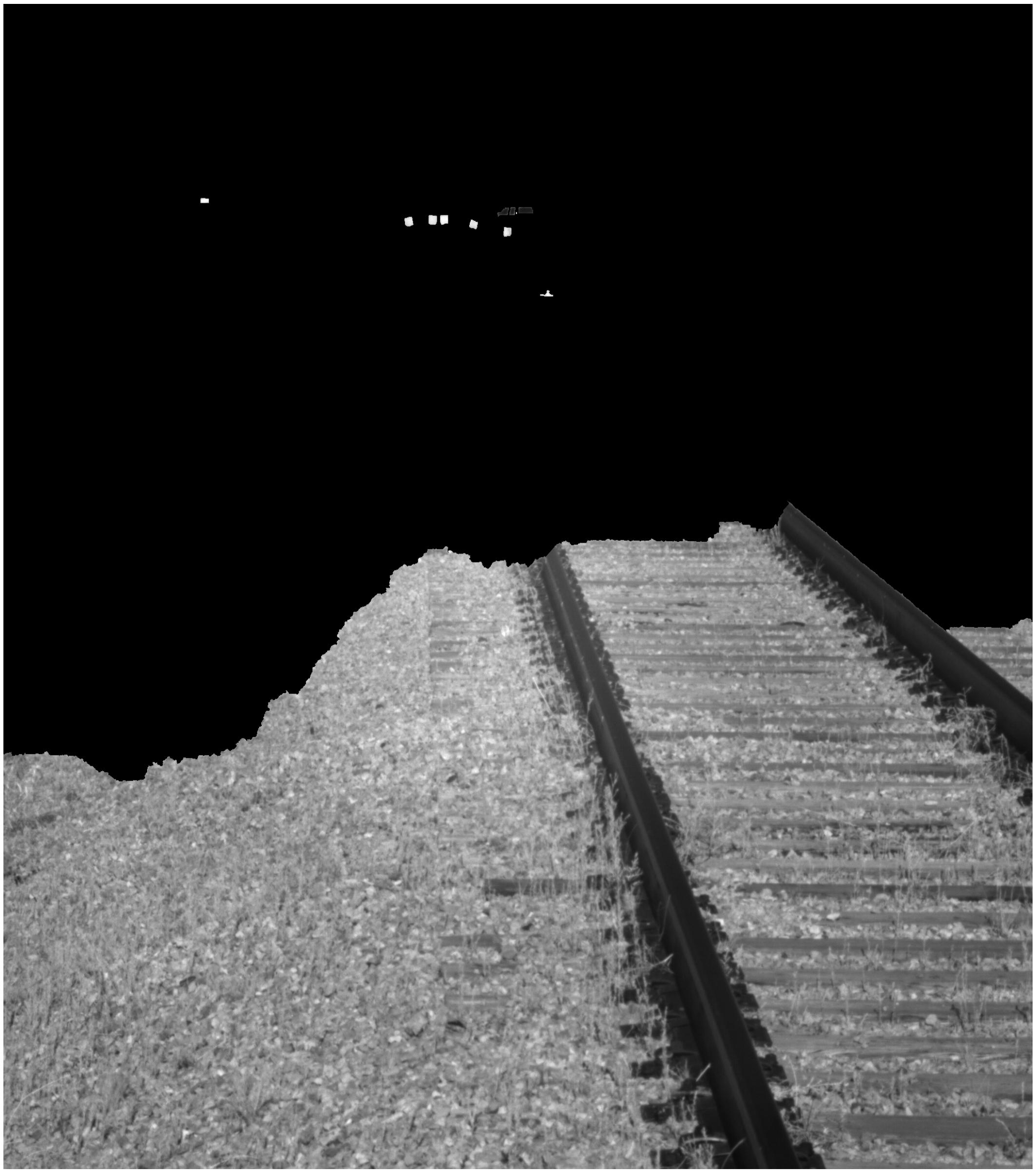


Figure A6. Foreground extracted from the input image for the spectral method of saliency using the GrabCut algorithm

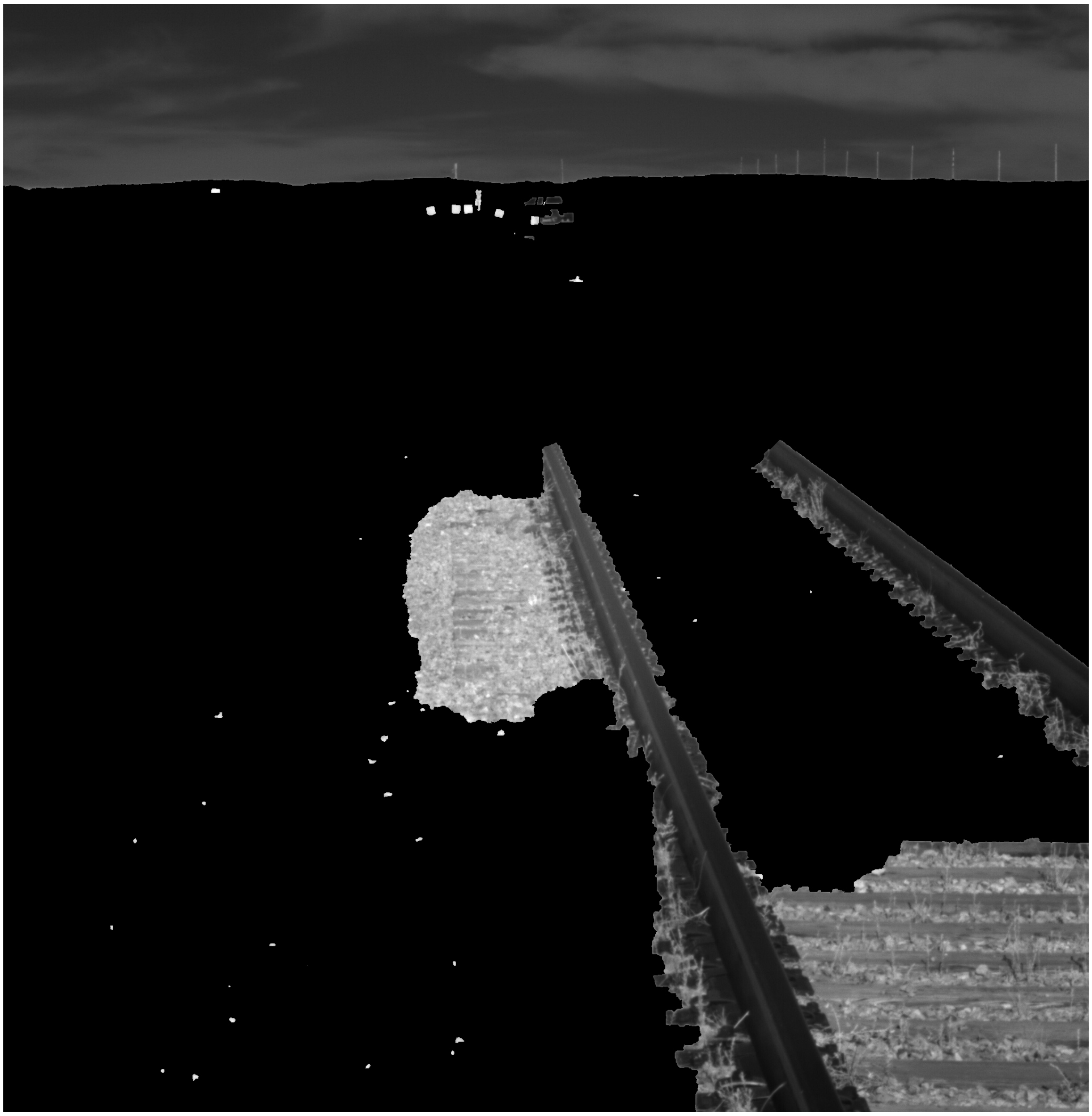


Figure A7. Foreground extracted from the input image for fine-grained method of saliency using the GrabCut algorithm

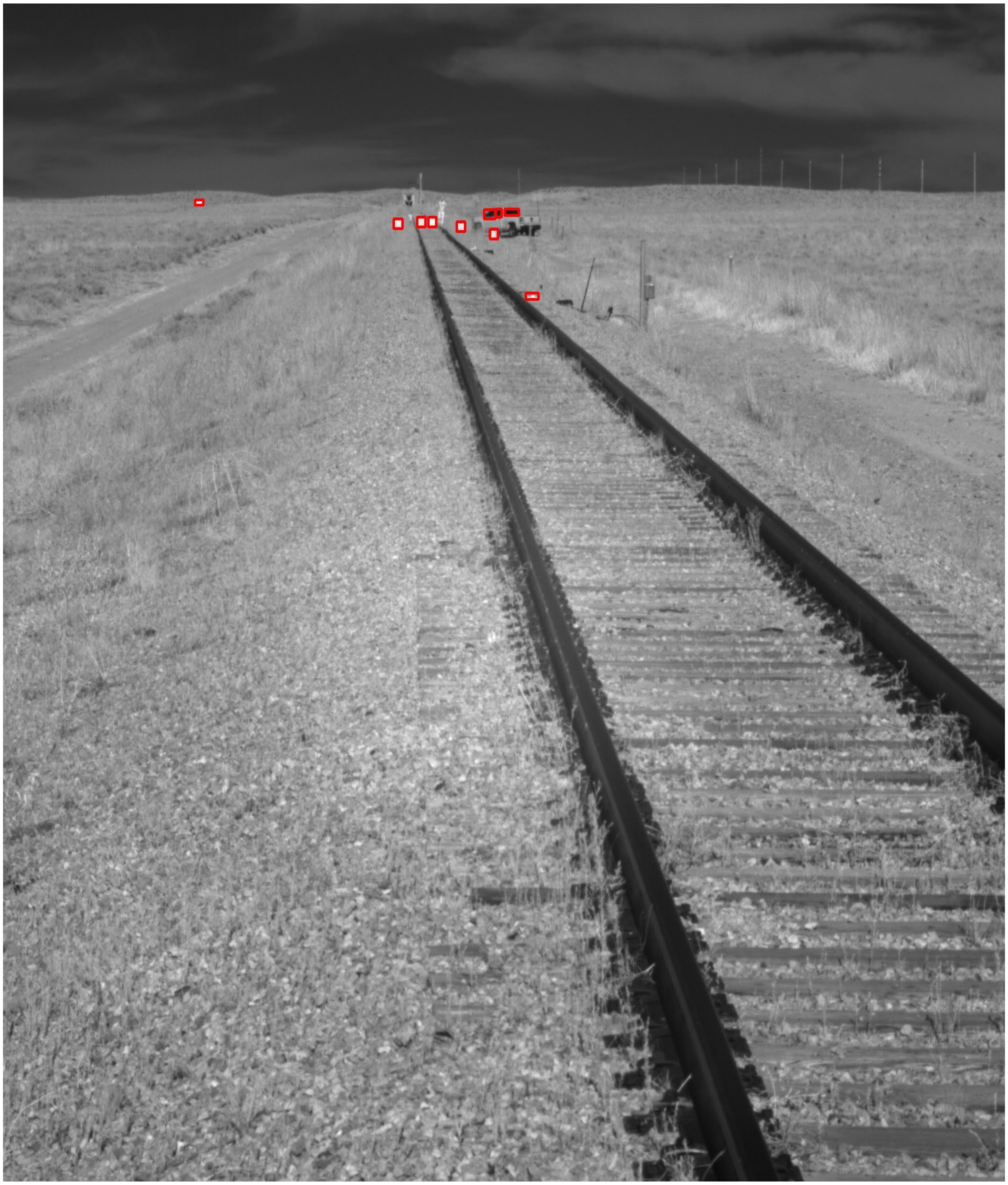


Figure A8. Spectral residual object localization

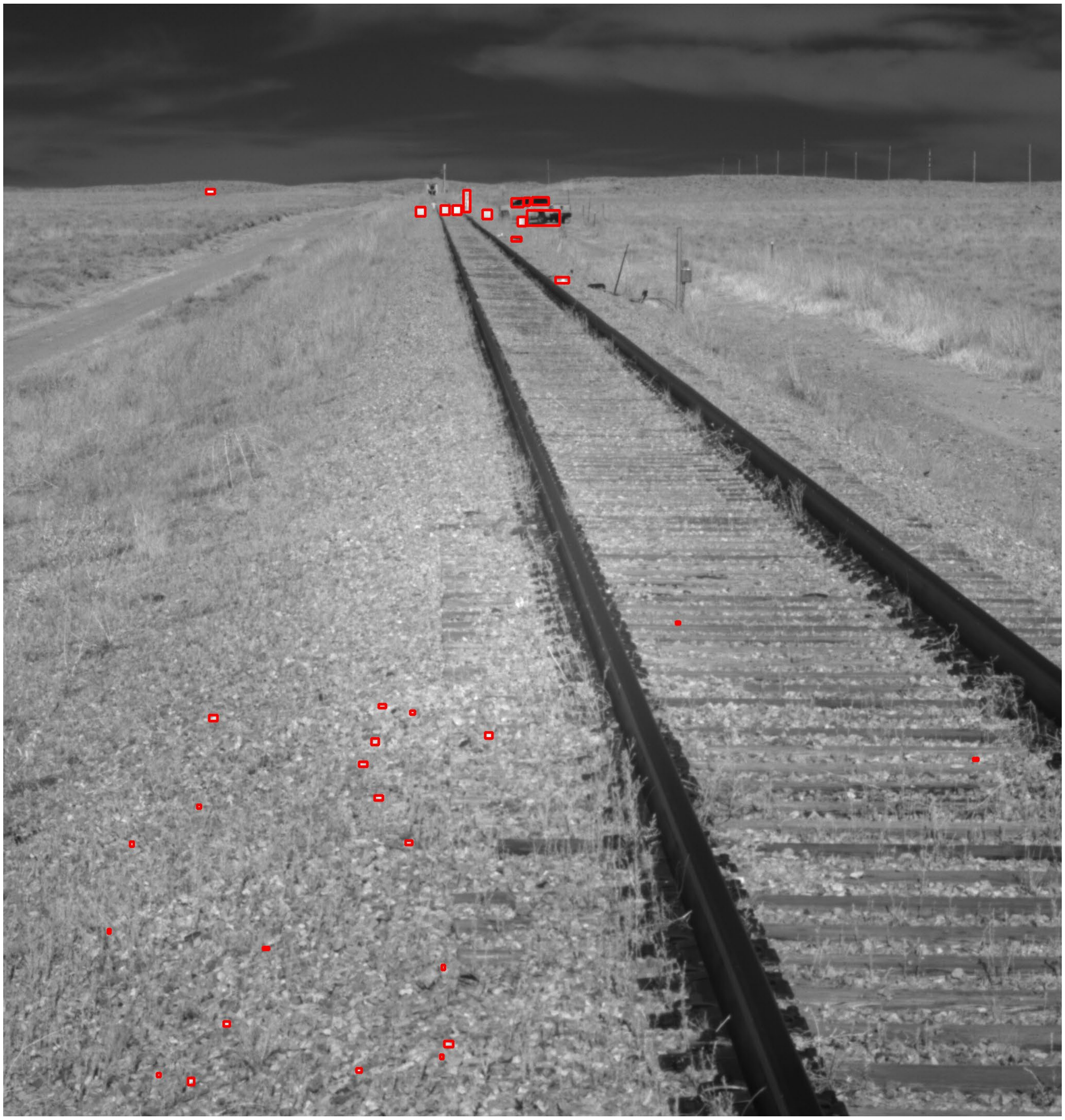


Figure A9. Fine grained object localization

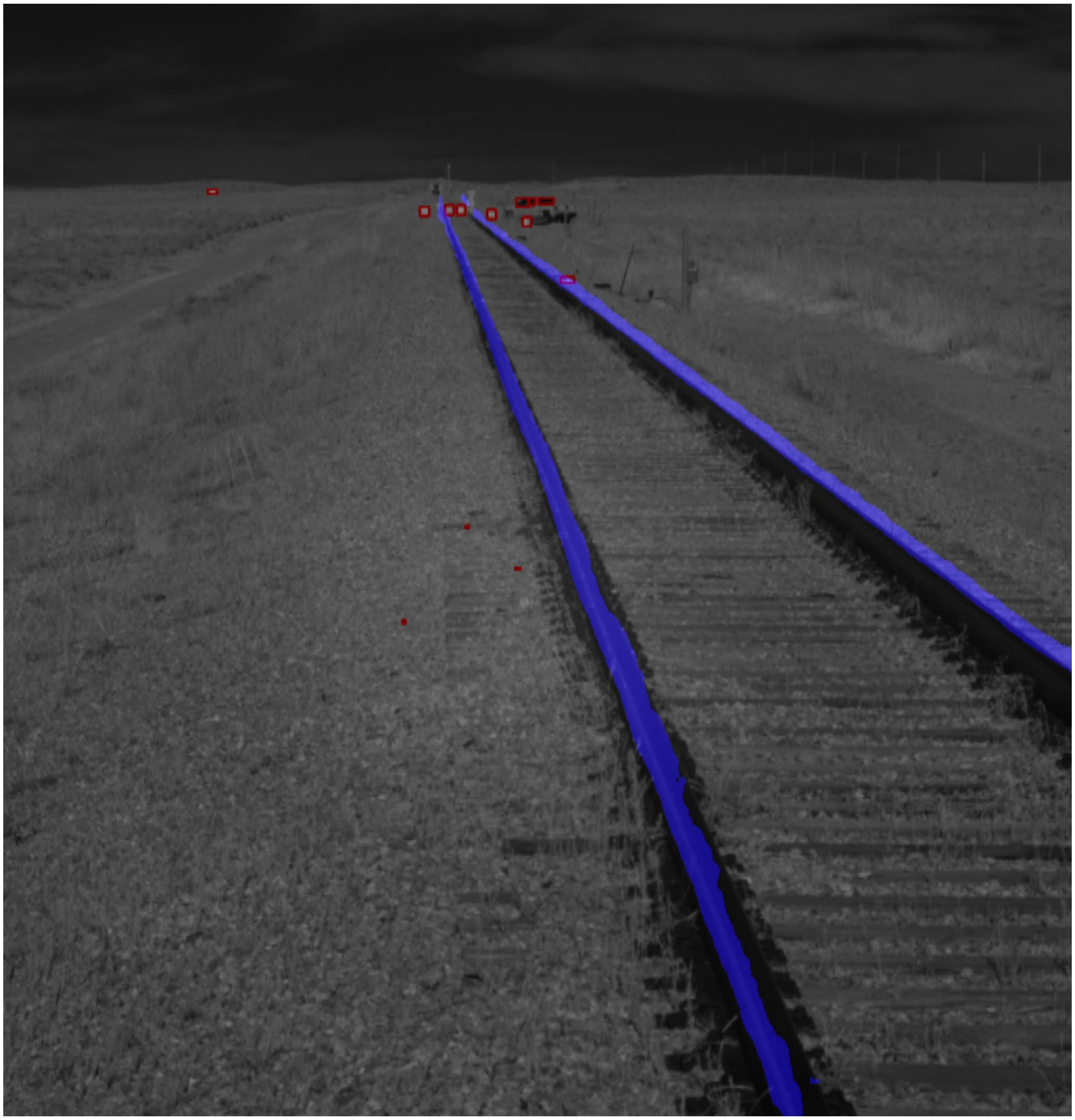


Figure A10. Fine grained object localization and rail segmentation

Abbreviations and Acronyms

ACRONYM	DEFINITION
AG	Advisory Group
ATO	Automated Train Operations
CNN	Convolutional Neural Network
COCO	Common Objects in Context
COI	Conditions of Interest
COTS	Commercial Off the Shelf
CSP	Cross Spatial Partial
DA	Data Analysis
DNN	Deep Neural Network
EMS	Energy Management Systems
FOV	Field of View
FRA	Federal Railroad Administration
GMM	Gaussian Mixture Models
GNU	GNU’s not Unix
GPU	Graphics Processing Unit
HOG	Histogram of Oriented Gradient
HOT	Head of Train
IOU	Intersection Over Union
ITC	Interoperable Train Control
OOI	Object of Interest
ReLU	Rectified Linear Unit
RFP	Request for Proposal
RP	Rapid Prototype
SP	Sensor Platform
TWG	Technical Working Group