Accelerating Rural Road Safety Using AI to Unlock Predictive Insights from Videolog Data

Final Report

Prepared By:

University of North Carolina Renaissance Computing Institute

Volpe National Transportation Systems Center

University of North Carolina Highway Safety Research Center

Prepared For:

North Carolina Department of Transportation

USDOT Safety Data Initiative

November 2021

Technical Report Documentation Page

1. Report No. RP2021-20	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Accelerating Rural Road Safety Using Artificial Intelligence to Unlock Predictive Insights from Videolog Data		5. Report Date November 2021 6. Performing Organization Code
7. Author(s) Hong Yi, Chris Bizon, David Borland, Matthew Watson, Matt Satusky, Robert Rittmuller, Randa Radwan, Raghavan Srinivasan, and Ashok Krishnamurthy		8. Performing Organization Report No.
9. Performing Organization Name and Address Renaissance Computing Institute (RENCI) (University of North Carolina), Volpe National Transportation Systems Center, and Highway Safety Research Center (University of North Carolina)		10. Work Unit No. (TRAIS) 11. Contract or Grant No. NCDOT RP2021-20 USDOT FAIN 69A34520501070620
12. Sponsoring Agency Name and Address US Department of Transportation 1200 New Jersey Ave, SE Washington, DC 20590	NC Department of Transportation Research & Development 1020 Birch Ridge Dr Raleigh, NC 27699	13. Type of Report and Period Covered Final Report August 1, 2020 – November 30, 2021 14. Sponsoring Agency Code
Supplementary Notes:		

16. Abstract

Roadway safety, especially in rural areas, is one of the most critical components in transportation planning. In collaboration with North Carolina Department of Transportation (NCDOT), UNC Highway Safety Research Center (HSRC), and DOT Volpe National Transportation Systems Center, UNC Renaissance Computing Institute (RENCI) developed a roadside feature detection solution leveraging multiple convolutional neural networks (CNNs). The solution uses an iterative active learning (AL) computer vision model training pipeline integrated into an AI tool to detect safety features such as guardrails and utility poles in geographically distributed NC rural roads. The RENCI team utilized transfer learning by adopting the Xception neural network architecture as the feature extraction backbone which was then used in an iterative AL process supported by a web-based annotation tool. The annotation tool not only allows for the collection of annotations through an iterative AL process for multiple safety features, but it also enables visual analysis and assessment of model prediction performance in the geospatial context. AL techniques were used to direct human annotators to label images that would most effectively improve the model aimed at minimizing the number of required training labels while maximizing the model's performance. The iterative AL process combined with a common feature extraction backbone allowed fast model inference on millions of images in the AL sampling space. This enabled a rapid transition between AL rounds while also reducing the computing requirements for each round. Model feature extraction weights were then fine-tuned in the last round of AL to obtain the best accuracy. Since only about 2.7% of 2.6 million unlabeled images in the AL sampling space contain guardrails, there is a significant class imbalance problem that had to be addressed in our AL sampling strategies for the guardrail classification model. Our AI tool can be used to detect roadside safety features and be extended to also locate them for assessing roadside hazards.

17. Key Words Artificial intelligence, Deep Learning, Convolutional Neural Networks, Transfer Learning, Active Learning, Class Imbalance, Roadside hazards, Rural roads, Videolog data, guardrail, utility poles	18. Distribution Statement		
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 10	22. Price

Form DOT F 1700.7 (8-72) **Reproduction of completed page authorized**

DISCLAIMER

Executive Summary

This project has developed an artificial intelligence (AI) tool for automated analysis, extraction, and annotation of roadside features on rural roads from existing video log data. The project was funded by the US Department of Transportation's Safety Data Initiative (SDI) through the North Carolina Department of Transportation (NCDOT). The developed AI tool uses a deep learning-based computer vision model to detect safety features such as guardrails and utility poles in geographically distributed NC rural roads with a high level of accuracy. Specifically, we applied the AI tool to two features, namely guardrail and utility pole. With only about 1.5% of 2.6 million unlabeled images annotated by domain experts, we achieved 99% accuracy for the guardrail classifier evaluated on a randomly selected holdout test set with over 12K images and 90% accuracy evaluated on a class-balanced holdout test subset with 642 total images. There were no class imbalance issues for the utility pole classification since about 64% of images in the AL sampling space contain utility poles. With only about 0.68% of 2.6 million unlabeled images annotated by domain experts, we obtained 88% accuracy for the utility pole classifier evaluated on a randomly selected holdout test set with 960 total images.

The AI tool used the Xception deep learning neural network architecture [1] to extract various features from the video log data which were then used in an iterative active learning (AL) computer vision model training process. The objective of AL is to minimize the number of instances that require human annotations while still producing a model with satisfactory performance, which can be achieved via effective sampling or query strategies. This training process was directly supported by a web-based annotation tool that enables rapid quality control and collection of end-user feedback throughout the data annotation process. More specifically, this annotation tool allows for the collection of annotations within each iteration of the AL process for multiple roadside features, while also enabling visual analysis and assessment of model prediction performance in the geospatial context. AL techniques were used to direct human annotators to label images that would most effectively improve the model aimed at minimizing the number of required training labels while maximizing the model's performance. The iterative AL process combined with a common feature extraction backbone allowed fast model inference on millions of images in the AL sampling space. This enabled a rapid transition between AL rounds while also reducing the computing requirements for each round. Specifically, this shared feature extraction backbone approach dramatically improved AL model prediction performance, reducing prediction time from about one week to under an hour. Model feature extraction weights were then fine-tuned in the last round of AL to obtain the best accuracy. Our AI tool can be used to detect roadside objects and be extended to also locate them for assessing roadside hazards. Ultimately, the capacity to accurately extract roadside features from video log data will help facilitate safety initiatives more efficiently and effectively on the state's rural roadway system. The AI annotation tool and visual geospatial analysis of model

predictions can serve as a blueprint that could be scaled and replicated for other jurisdictions and states.

Background

A key safety performance challenge in both North Carolina (NC) and other states is that severe and fatal injury crashes on rural roads are often dispersed across many miles in the roadway system. The current process for assessing roadside hazards on NC rural roads is site specific and is performed by individual field investigation per location, which is tedious and overtly limited given the large network of 48,673 miles of rural roads.

The goals for this project were to explore what roadside features could be reliably collected from videolog utilizing AI methodologies and techniques. The AI process could potentially allow NCDOT to assess thousands of miles of rural roadside features with minimal staff time. Example roadside features include continuous roadside objects such as guardrail, fences, and walls and point roadside objects such as utility poles, guardrail ends, and trees. An inventory of where these roadside features exist could allow NCDOT to better assess roadside risk along their roads when an errant vehicle leaves the roadway. NCDOT does not currently have an inventory of roadside features across the network. In addition to where roadside features exist along a network, distance to a feature and side slope to a feature are also important attributes that NCDOT does not currently have. The intent of the safety tool developed in this project is to enable the identification of roadside features at a systems wide level through an automated process, enabling a better understanding of the current road features on an expansive rural network.

In 2018 and 2019, NCDOT collected video log data for all secondary roads, 76 percent of which are rural roads. Images in the videolog were acquired by three front facing cameras every 26 feet along the roadways; an example set of three images can be seen in Figure 1. This wealth of data has provided NCDOT a unique and timely opportunity to partner with University of North Carolina (UNC) Highway Safety Research Center (HSRC) and Renaissance Computing Institute (RENCI) to develop an AI annotation tool to extract roadside features from the collected video log data to facilitate the determination of the safety level of the roadside across a large network of rural roads. This capacity will facilitate the identification of systemic safety initiatives to reduce fatal and serious injury crashes involving the roadside.

Figure 1. Example Annotation of a Guardrail from the Annotation Tool

For this one-year effort, the AI tool first allowed data analysts and safety researchers to semi-automatically annotate video logs with features of relevance to roadside safety. Using a web-based visualization tool, it employed deep learning-based computer vision methods to identify these features in unlabeled video, using an active learning feedback loop to rapidly direct human annotators to the most valuable segment of video for labeling, greatly speeding up the annotation process. The effectiveness of this approach was outlined in this report and was illustrative of how an iterative approach to developing these models could be key to reaching successful levels of performance. The active learning feedback loop helped to improve the ability of the AI tool to identify the selected features, and the additional manual effort would go from annotation to correction as the machine learning improved. This feedback loop would continue until the AI tool was able to reliably identify features automatically.

Methodological Approach

We used an iterative process to create a training data set and a predictive image classification model to extract safety-related roadside features from video log data. Figure 2 below shows our iterative process, and is further described below

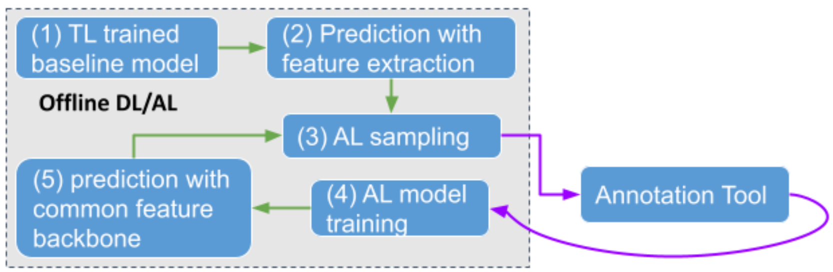


Figure 2. Our AI Tool Iterative Processing Pipeline

First, we trained a baseline model using Xception architecture [1] via Transfer Learning (TL) by leveraging the primary road guardrail assessment data NCDOT collected in 2017 which was manually extracted from the primary road video log data and provided valuable information for us to prepare labeled images for training; second, we used this baseline model to extract features in a common feature extraction backbone and make predictions for millions of unlabeled secondary road images in the AL sampling space; third, we computed most informative samples for AL based on model predictions and extracted features; fourth, these samples were then fed into the annotation tool to collect expert annotations for these selected sample images; fifth, manual annotations were exported from the annotation tool and used to train a new model as part of AL; sixth, the new model was used to make fast predictions with a common feature extraction backbone using the extracted features from the baseline model, and the new model predictions were fed into AL sampling in step 3, and the AL loop continued until the new model reached our target performance metrics. Last, the weights in the final AL round model including the feature extraction backbone were fine-tuned to obtain the best model accuracy. Through this AL process, we achieved 90% accuracy for the final guardrail model starting from the initial 75% accuracy for the initial baseline model and the 88% accuracy for the final utility pole model starting from the initial 66% accuracy for the initial model evaluated on a balanced holdout test set. We describe our AI tool processing pipeline and methodology in detail next.

Baseline Model Training via Transfer Learning

We used the Xception [1] network pre-trained on ImageNet to train guardrail models on the whole data set and the two-lane only subset via TL. Xception was a deep CNN architecture from Google inspired by its predecessor Inception, where Inception modules have been replaced with depthwise separable convolutions. Xception architecture has the same number of parameters as Inception V3 but has better performance than Inception V3 due to a more efficient use of model parameters. Chollet [1] showed that Xception slightly outperformed Inception V3 on the ImageNet dataset which Inception V3 was designed for, and significantly outperformed Inception V3 on a larger image classification dataset comprising 350 million images and 17,000 classes.

We used TensorFlow (TF) [2] 2 as the Deep Learning (DL) framework which includes Xception architecture with pre-trained weights that can be directly loaded in our application. We replaced the top layer of the Xception with three stacked layers: a Global Average Pooling layer to reduce the number of parameters by 100-fold, followed by a dropout layer with a rate of 0.25 to help prevent overfitting, and a fully connected layer on the top with a sigmoid activation for binary classification with the binary cross-entropy as the loss function. We used an Adam [3] optimizer with a learning rate of 0.001 initially to train the top classification layer only for 10 epochs which improved the model accuracy from the initial 54% to about 91%. Then we kept the bottom two block layers frozen and opened up the top 12 block layers of Xception for fine tuning the pretrained weights with a very low learning rate of 1e-5. In addition, we used the model checkpoint callback supported by TF to monitor validation loss during training to only save the best performing model with minimal validation loss at the end of each epoch. Figure 3 shows the accuracy and loss plots of the guardrail models trained on full data and 2-lane only subset data during fine tuning, from which we can see that the full data model started overfitting after epoch 7 (top) and the 2-lane only model started overfitting after epoch 10 (bottom). We selected epoch 7 full data model and epoch 10 2-lane only model with minimal validation losses as the baseline model candidates. Overfitting occurs when the model fits too well against its training data but does not fit well against its validation data. When overfitting happens, the model will not generalize or perform well against new unseen data for prediction. To make sure that our model converged, we evaluated the model on validation data at the end of each epoch to make sure the model we selected did not overfit.

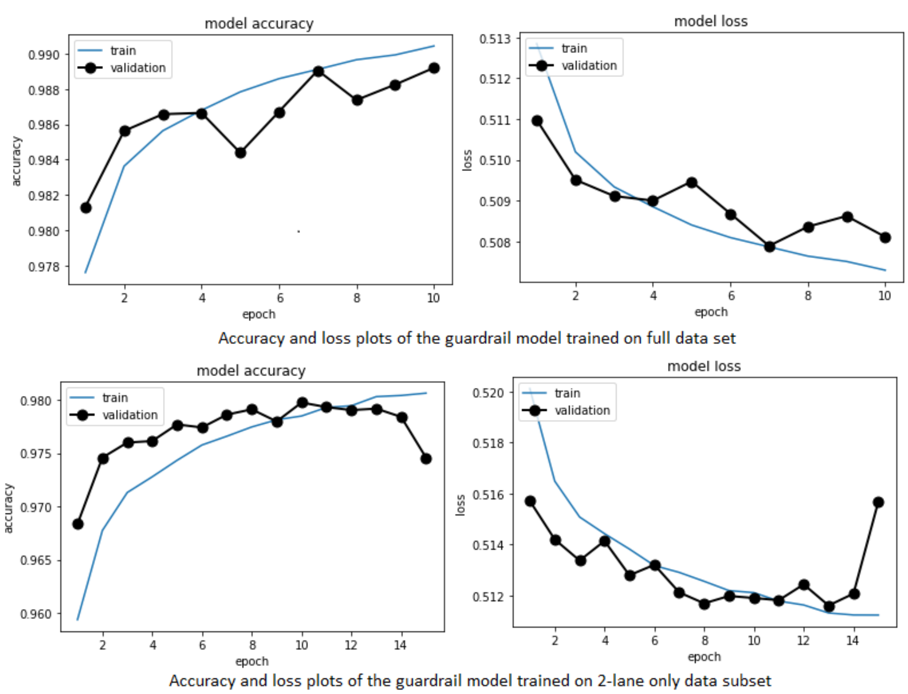


Figure 3. Accuracy and Loss Plots of the Guardrail Models Trained on Full Data (Top) and 2-Lane Only Data (Bottom) During Fine Tuning

We evaluated two model candidates on the 2-lane only balanced primary road test set which aligned more with the secondary road target set. Two model candidates had similar performance although the 2-lane only model has a little better accuracy 97.9% than 97.7% of the full data model. In addition, we compared predictions of two models on a NC eastern region of 741,476 joined images in the secondary road target set and found two models have the same predictions for 98% of all images. We manually inspected a subset of 300 randomly selected images and compared the receiver operating characteristic (ROC) area under curve (AUC) scores of both model predictions on the subset, with 100 out of the total image set, 100 out of the 98% common prediction image set, and the other 100 out of the 2% different prediction image set. Both manual inspection and AUC scores indicated the 2-lane data model performed a little better (0.903 AUC score for the 2-lane model vs. 0.853 for the full data model). As a result, we selected the 2-lane data model as our baseline guardrail model for the subsequent AL.

Common Feature Extraction Backbone for AL

Our objective was to create safety feature models via AL to make predictions in our unlabeled secondary 2-lane rural road target set (over 40 thousand miles) composed of 14 divisions across NC. NCDOT transportation safety experts selected 4 geographically representative divisions in the target set that we used to create the AL sampling image space. There are about 2.6 million joined image samples in the AL sampling space from which we selected the most informative samples for manual labeling based on model predictions from the previous AL round aimed at achieving high model accuracy using as few labeled instances as possible.

To ensure a smooth transition between AL rounds, predictions of the model from one AL round over all 2.6 million joined images or 7.8 million single view images in the AL sampling space must be finished fast enough to start the next AL round in a reasonable amount of time. To address this AL performance issue, we employed a common feature extraction backbone approach through the iterative AL process. In particular, feature extraction from input images was performed by a common backbone network in a single pass and shared across AL rounds. Our common feature extraction backbone consisted of a set of convolution and polling layers in the Xception architecture to produce a feature map containing higher-level summary information of images. Throughout the AL rounds, only the top fully connected classification layer was trained with 2048 input parameters from the shared common backbone network and one output parameter for binary classification with sigmoid activation. This shared feature extraction backbone approach dramatically improved AL model prediction performance, reducing prediction time from about one week to under one hour.

In addition, having a shared common feature extraction backbone across the AL rounds allowed for methods such as similarity-based sampling for effective AL sampling under class imbalance via analysis of similarities of extracted image feature vectors in the feature embedding space.

Annotation Tool for Supporting AL

We developed a web-based tool, Roadway Hazard Finder (RHF), to collect annotations from transportation safety experts to support AL. As a central component of our AI system, RHF was tightly integrated with our offline DL/AL pipeline, enabling easy ingestion of image samples selected from offline AL into the tool for annotation, and uploading of collected annotations from the tool for offline AL model training. The primary annotation interface receives images for annotation prioritized by the AL system and provides a simple point and click interface for annotating the images based on the current feature of interest. The annotation tool also provides diagnostic interfaces, such as a route browser (Figure 4) that enables the user to virtually drive a particular route while visualizing a plot of all predictions and annotations along the route, and a prediction errors table (Figure 5) that enables the user to organize and review any discrepancies between the model predictions and user annotations.

The annotation tool was designed to be versatile and easily re-purposed for other similar tasks. For example, we used the annotation tool internally to prepare the holdout test sets for objectively assessing guardrail and utility pole models through the AL process. Due to the significant class imbalance for the guardrail feature (only about 2.7% of 2.6 million unlabeled images in the AL sampling space contain guardrails), we annotated 12,057 randomly selected images across NC eastern, central, and western regions in our AL sampling space in order to collect 321 positive images with approximately even distributions across the three regions. Given the class imbalance in this whole holdout test set, we also randomly selected 321 negative images out of the 11,736 total to create a balanced holdout test set to objectively assess guardrail model performance. On the other hand, the utility pole data classifications do not exhibit class imbalance. As a result, we annotated 960 randomly selected images across three representative NC regions and collected 613 positive images and 347 negative images as the pole holdout test set for assessing performance of AL pole models.

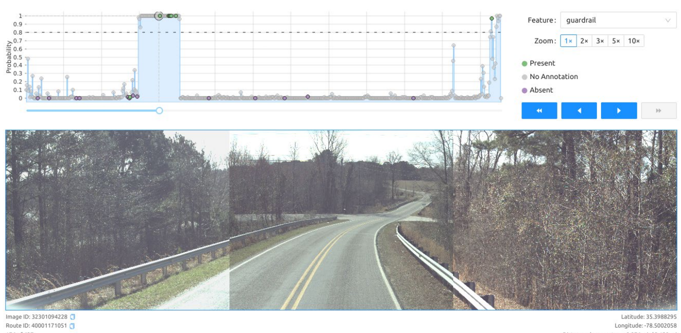
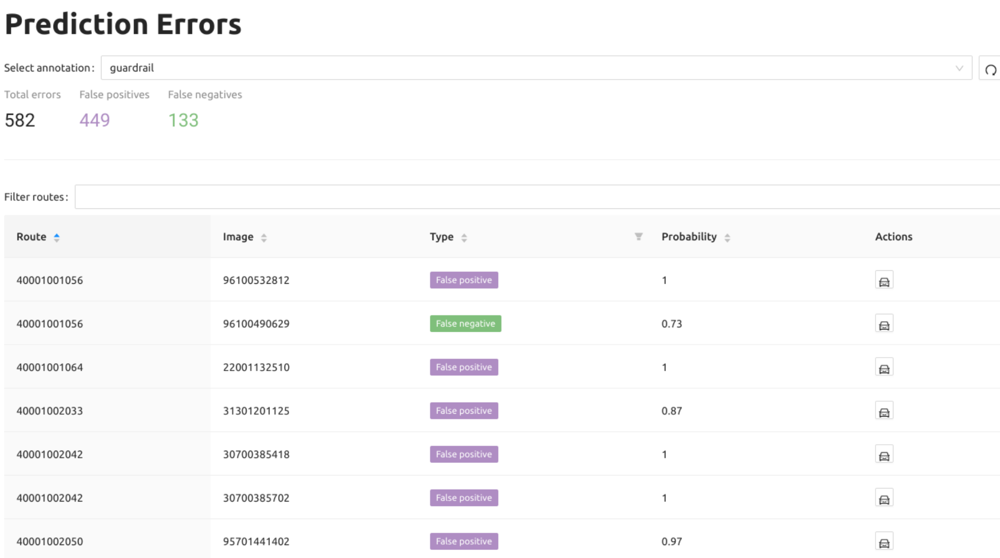


Figure 4. Route Browser with Scatterplot Depicting Predictions and Annotations



Prediction Errors

Select annotation: guardrail

Total errors	False positives	False negatives
582	449	133

Filter routes:

Route	Image	Type	Probability	Actions
40001001056	96100532812	False positive	1	
40001001056	96100490629	False negative	0.73	
40001001064	22001132510	False positive	1	
40001002033	31301201125	False positive	0.87	
40001002042	30700385418	False positive	1	
40001002042	30700385702	False positive	1	
40001002050	95701441402	False positive	0.97	

Figure 5. Prediction Errors Table

Active Learning Sampling Strategies

Our goal for AL sampling strategies was to sort images for human annotation in the order of likelihood that the manually labeled images would contribute most to improve the model performance. Specifically, an offline process computed an uncertainty score for each image sample that corresponded to its sampling order determined by an AL sampling strategy. These uncertainty scores were then ingested into the annotation tool and used to return images for annotation in the order intended by the sampling process. We used the baseline model as described in the earlier section Baseline Model Training via Transfer Learning to predict guardrail probabilities of unlabeled images in our AL sampling space before we started the AL process.

The objective of AL is to minimize the number of instances that require human annotations while still producing a model with satisfactory performance, which can be achieved via effective sampling or query strategies. Most AL approaches employ the uncertainty sampling strategy [4] to select data along the decision boundary where the model is most uncertain for labeling. However, these instances along the decision boundary may not be representative of the unlabeled data pool and may sometimes lead to the selection of unrepresentative outliers or too many common class instances in data with significant class imbalance. To address these issues, some other heterogeneity-based sampling strategies may be employed to sample from instances that are dissimilar to what has already been sampled [5]. For example, the query-by-committee sampling approach [6] uses a committee of different classifiers to predict the class label of each unlabeled instance and selects those instances for which the classifiers disagree most. Different strategies have different trade-offs depending upon the underlying application and data distribution. We employed uncertainty sampling and the query-by-committee strategies to fit better with our application and data distribution in conjunction with a similarity-based sampling strategy we developed by adapting ideas proposed in the similarity-based AL framework [7] to address the significant class imbalance problem for guardrail classification. Specifically, we developed a similarity-based sampling strategy by adapting the rare class and dissimilar sample selection ideas proposed in the SAL framework to take into account the extreme class imbalance for guardrail classification. Similar to the SAL framework, our sampling strategy attempted to sample more rare class samples effectively as well as images dissimilar from those already in the training dataset by sorting similarity scores computed in the feature embedding space. However, this algorithm resulted in many noisy outlier images being selected which did not represent the real distribution of the new data. We applied some basic image processing techniques to filter out those noisy outlier images with extreme over- or under-exposure to capture those unlabeled images of high entropy (most uncertainty) and diversity with less chance of selecting trashy images.

AL Model Training

We used the Xception feature extraction backbone from the baseline guardrail model throughout the iterative AL model training process. Specifically, only the top fully connected classification layer with 2049 parameters were trained through the AL rounds with the weights in the shared common feature extraction backbone frozen. We randomized initial weights from a uniform distribution and reset biases to zeros for the top classification layer before training for each AL round which we found was critical for good model convergence.

The baseline guardrail model was trained on joined images of left, front, and right views since the guardrail assessment data we used to prepare training data could only be mapped to joined images by geographic locations. On the other hand, our captured labeled data include annotations for each single view image, enabling us to train AL models on single view images. Since we resized our input images with varying resolutions to the original input image size 299x299 for the Xception architecture, training on joined images would have threefold increase in likelihood that some thin, short, faraway features could disappear due to image resizing, which would increase FNs in model predictions and challenge models with the conflicting information between the annotations and resized images fed to models. As a result, we used collected annotations to prepare single view images for training AL models.

At each AL round, we trained the model using all annotations collected so far randomly split into 80% for training and 20% for validation, which created unbalanced training and validation data. Some commonly used approaches such as over-sampling rare class or under-sampling common class instances change the class distributions and don't work well for our application. Instead, we adjusted class weights that each training instance carried when computing the loss by giving rare class instances more weight than the common class instances. We also used early stopping callback supported by TF to stop training when the validation loss did not improve for 10 epochs.

Final Model Weight Fine Tuning

Although the common feature extraction backbone was used through AL to allow fast model training and inference through the AL process enabling quick transition between AL rounds, it did set an upper limit for final model accuracy due to less than ideal weights for the common backbone. Specifically, since the common backbone was extracted from the baseline guardrail model which was trained on primary road joined images, the weights in the common backbone may not be satisfactory for the secondary road single view image target set. By fine tuning weights for the common feature extraction backbone for the final round model of AL, we improved the guardrail model accuracy from 84% to 90% evaluated on the balanced holdout test set and the pole model accuracy from 72% to 88%. Although fine tuning weights for the final model was justified by the big model accuracy improvement, a better designed common feature extraction backbone might be beneficial and remove the need for weight fine tuning.

Results

Figure 6 and Figure 7 show ROC curves of the guardrail and utility pole models through AL evaluated on the respective balanced holdout test sets. As shown in the ROC curves, the final round of models with fine-tuned feature extraction weights had significant performance improvement over the AL models trained with the shared common feature extraction backbone. In addition, there was a big performance improvement in the first AL round guardrail model over the baseline model followed by minor performance improvement in subsequent AL rounds. Similarly, there was minor model performance improvement over AL rounds for the pole model.

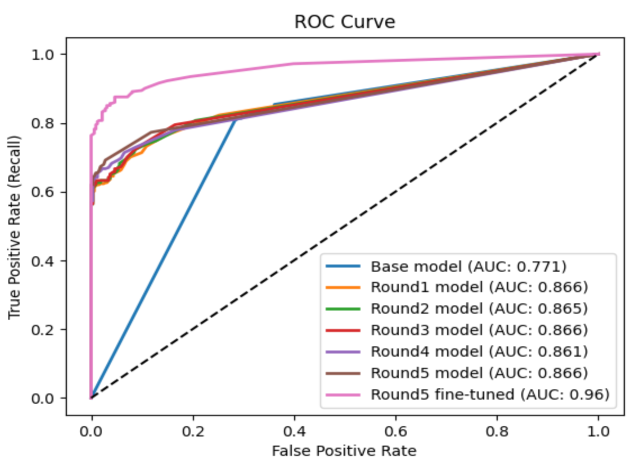


Figure 6. ROC Curve of Guardrail Models through AL

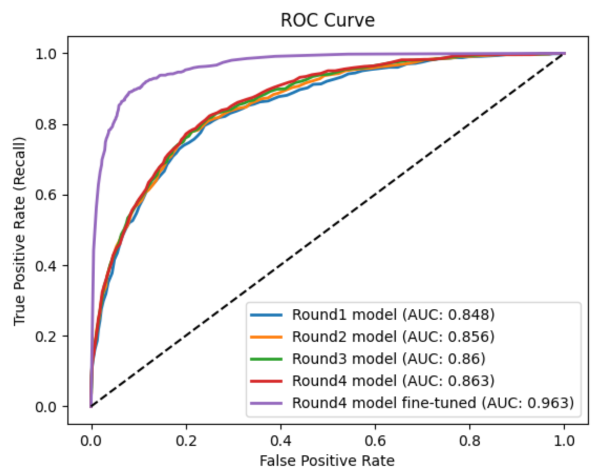


Figure 7. ROC Curve of Pole Models Through AL

Conclusions

Lessons Learned

- **AI can identify video frames containing safety-related roadside features:** As shown above, AI image classification models are able to identify images containing roadside safety-related features with high accuracy. The models developed include both extended features (guardrails) and localized features (utility poles). Furthermore, thresholds can be tuned to favor either recall or precision to aid in particular use cases. The basic procedure, models, code, and tools are all fully extensible to both new features and may be deployed in other states.
- **A common backbone approach works well for training but must be relaxed to produce the best models:** A novel feature of the current work is the use of a common computational backbone across models for different features. This exploits the fact that many computer vision models require similar decompositions of an image, which are then combined in specific ways to find specific features. The significance here is that use of this common backbone allows large portions of the model to be calculated once and then subsequently re-used, vastly reducing the computational time required to make predictions and therefore allowing a much faster turnaround time between human labeling of images and AI predictions of roadside features. However, while the efficiency gains are critical in rapidly training the models, to achieve the highest accuracy

for an individual feature, we found that the common backbone must be allowed to slightly evolve as a fine-tuning step.

- **A combined team of roadway safety experts and AI researchers is integral to rapidly producing high-accuracy models:** The success of this project was due to a team composed of many different skills and backgrounds including NCDOT safety professionals, safety researchers from HSRC, and AI and UI practitioners from RENCI and Volpe. This particular set of talents was critical in creating working tools in short order, drawing critically on the background of NCDOT personnel to define goals and provide insight into relevant problems and AI researchers to translate those goals into working tools and models.
- **Effectively integrating AI results into State DOT workflows will require moving beyond image classification:** The image classification approach taken in this work is a low-cost approach to produce high-accuracy models relevant to NCDOT needs. However, the limitations inherent in this approach do limit its domain of applicability. Image classification only identifies that an image contains e.g., a utility pole. It does not describe where in the image the pole exists, nor does it describe that location in real world coordinates; poles near the road are identified, as are poles that are far from the road, and are therefore not relevant for lane-departure events. In some circumstances, such as guardrails, the infrequency of the feature coupled with their paradigmatic location in images mitigates these problems, but for more frequent point features such as poles, being able to place the objects in space is critical for application of the models to actual DOT problems.
- **Project AI results integrated into State DOT roadway network:** NCDOT was able to incorporate the project results for the continuous roadside object of guardrail and the point roadside object of utility pole onto the State GIS linear referencing system (Figure 8 shows the NCDOT's linear referencing system with location of guardrail; Figure 9 shows the location of utility poles). These information layers could be considered the first step in assessing network wide roadside risk. As mentioned in the previous bullet, videolog images screened through the safety tool can produce where guardrails and utility pole exist from the image perspective (i.e., "yes" or "no" for the distance of the image with respect to the distance the AI can view within the image). However, additional details such as exactly where a guardrail begins and ends and where a utility pole is exactly located can be future growth opportunities to this tool. The integration of AI methodologies to extract longitudinal and horizontal distance to a feature and side slope to a feature could improve upon the tool built in this project. In addition, future iterations of the tool could be developed to include more roadside continuous and point features for a more complete roadside inventory. A robust roadside feature inventory coupled with historical crash data could allow NCDOT to better assess potential risks along the large network of rural two-lane roads, and the prioritization of systemic safety countermeasure applications with limited safety dollar resources.

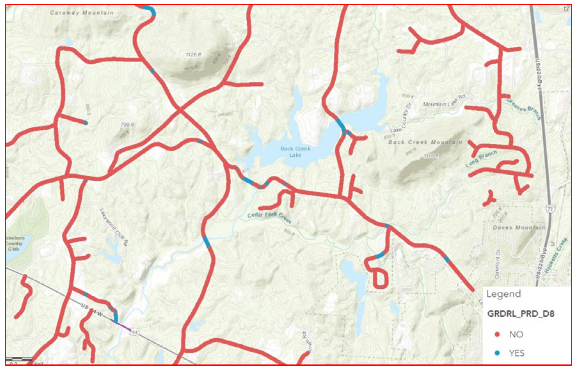


Figure 8. GIS Linear Referencing System with Guardrail Location

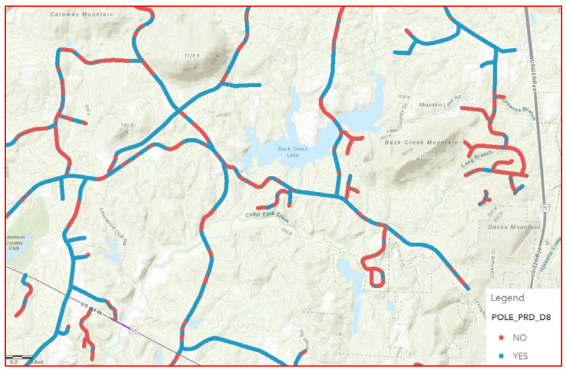


Figure 9. GIS Linear Referencing System with Location of Utility Poles

- **The rate-limiting resource is labeled data:** Building accurate models requires data to train the models, meaning images that a human has labeled as containing or not containing a specific feature of interest. Because it requires a person to view an image and, in some way, mark it, this process can be slow, but the amount of such data is one of the main factors in producing an accurate model. To mitigate this difficulty, we designed a UI system to allow users to efficiently annotate as many images as possible, and we used an active learning approach to choose the most influential images to label. While the combination of these tools minimized the amount of human time required for image labeling, this labeling still represented the main bottleneck.
- **Documentation and Information:** The supporting documentation and information can be found at the following link: https://github.com/RENCI/ncdot-road-safety

Further Work

Extensions to the current work can focus on addressing the limitations noted above. Specifically, we see three straightforward paths to make the current work more directly useful to state DOT personnel. First, we suggest using a combination of AI and classical computer vision algorithms to move beyond the image classification paradigm and place located features directly into three-dimensional space. Second, recent advances in self-supervised learning can be employed to further reduce the number of images that must be hand-labeled. Finally, the incorporation of other data streams, such as NCDOT created airborne LIDAR data will allow a data fusion approach in which LIDAR can be used to extend models of interest, as well as providing information that is difficult to extract from the video data alone, such as ground topography.

References

[1] F. Chollet, "Xception: deep learning with depthwise separable convolutions," IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), vol. 1, 2017, pp. 1800–1807.
[2] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, et al. "Tensorflow: a system for large-scale machine learning," Proceedings of the 12th USENIX Conf. on Operating Systems Design and Implementation, 2016, pp. 265–283.
[3] D. Kingma and J. Ba, "Adam: a method for stochastic optimization," 3rd International Conf. on Learning Representations (ICLR), 2015, abs/1412.6980.
[4] D. Lewis and J. Catlett, "Heterogeneous uncertainty sampling for supervised learning," 11th International conf. on machine learning (ICML), 1994, pp. 148–156.
[5] C. Aggarwal, X. Kong, Q. Gu, J. Han, and P. Yu, "Active learning: a survey," Data Classification: Algorithms and Applications, CRC Press, 2014.
[6] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," ACM Workshop on Computational Learning Theory, 1992, pp. 287–294.
[7] C. Zhang, W. Tavanapong, G. Kijkul, J. Wong, P. Groen, and J Oh, "Similarity-based active learning for image classification under class imbalance," IEEE International Conf. on Data Mining, 2018, pp. 1422–1427.