HIGH SCHOOL EDITION
Journal of Student Research
Volume 13 Issue 3 (2024)

Evaluation of Object Detection Capabilities of Autonomous Vehicles in Harsh Weather Conditions

Pranay Ranjan[1] and Noah Curran[#]

[1]Richmond Hill High School, Canada
[#]Advisor

ABSTRACT

In the last 20 years, we have witnessed significant advancements in the research on autonomous vehicles (AVs), but challenges remain in their global usage and adoption. According to government data, most AV testing in North America occurs in the desert areas of the southern US where weather challenges are minimal. Little test data is available from areas which encounter harsh weather (e.g., snow, heavy rain, dense fog). Given that object detection is a critical functional requirement for AVs, in this paper, we examine the effectiveness of object detection capabilities of AVs in harsh weather. The first part of our hypothesis is that industry-leading models, mostly trained on datasets from normal weather conditions, will not perform well in detecting objects in harsh weather. The second part of our hypothesis is that object detection accuracy should improve once these models are trained on harsh weather datasets. We used two industry leading models (Faster R-CNN and SSD) with KITTI (normal weather) and CADC (harsh weather) datasets for our research. Our experiment shows that when trained with KITTI data, both models had an extremely low object detection precision on the CADC dataset, supporting the first part of our hypothesis. In addition, our experiment also shows significantly improved detection precision when these models were trained on the harsh weather dataset, thereby proving the second part of our hypothesis. These results are valuable suggestions to the AV industry as it works to expand the deployment of AVs to harsh weather areas in the future.

Introduction

The field of Autonomous Vehicles (AVs) has garnered considerable attention from the public and particularly those in the fields of engineering and computer science. AVs have evolved with the addition of different controls and features, leading to the deployment of level-2 (partial driving automation) to level-4 (high driving automation) capable vehicles in many commercial applications. While significant progress has been made, government issued permits for the deployment of AVs currently limit their use under harsh weather conditions. This limitation underscores the existing gap in technology and the reliability of AVs in such challenging driving conditions. Also, as per government data, most of the testing of AVs in North America on public streets is still being conducted in the southern US where weather challenges are minimal (Figure 1).

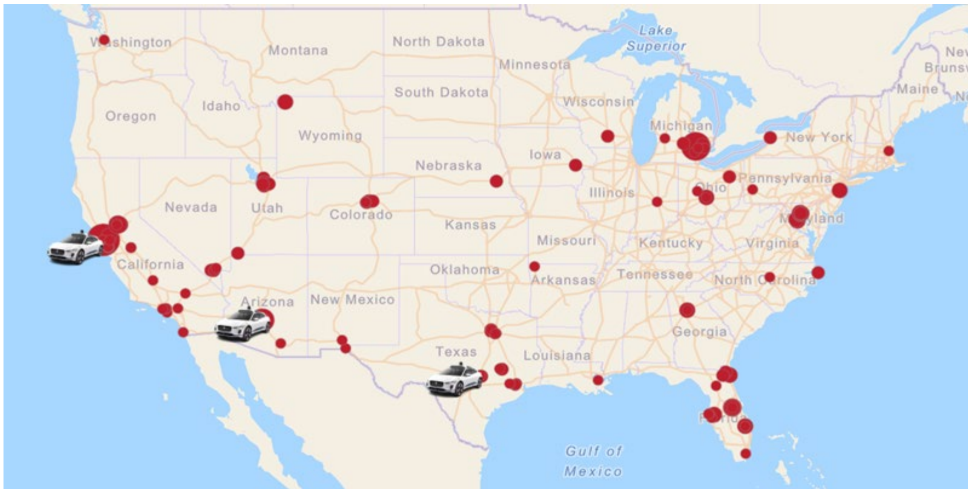

Figure 1. AV Test locations in USA. Bigger red dots represent locations where companies have reported higher number of vehicles and among these locations, testing on public streets is carried out only in California, Arizona, and Florida. Source is National Highway Traffic Safety Administration.

AVs have a pipeline of technical components that carry out specific tasks and interact with each other while driving. These components can be summarized as perception, planning, and control related. The inputs from raw sensors (camera, LIDAR, RADAR, ultrasonic sensor, etc.) are fed to the perception components to assess the environment and vehicle state. Based on these inputs, prediction and planning tasks are carried out and are eventually used by control components to generate motor commands. Within this technology stack, perception components play the most important role in the performance of AVs in harsh weather conditions, ensuring accurate detection of weather conditions, traffic objects, as well as the location of lanes and surface condition of the roads. Computer vision technology has been the main driver in the advancement of the effectiveness of object detection in AVs. The object detector should be able to accurately classify traffic objects (cars, pedestrians, etc.) and localize those objects by drawing bounding boxes around them. Early object detection models were built as a collection of hand-crafted feature extractors such as Viola-Jones detector and Histogram of Oriented Gradients (HOG). However, these models were not effective in dealing with the detection challenges of different types of objects (classes) at real time speed. The Convolutional Neural Network (CNN) based deep learning algorithms addressed most of these challenges by eliminating the need for hand-crafted feature engineering and having the ability to automatically learn hierarchical features by taking image as an input and computing its features at different layers of extraction. After AlexNet's demonstration of superior performance in ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2022, deep learning-based object detection algorithms have become commonly used for object detection tasks in AV and have been an active area of research in the last several years.

Deep learning-based object detection holds great promise for advancing AVs, but it requires training on robust datasets to handle the complexity of real-world driving scenarios. The quality, quantity, and diversity of the data used for training, validation, and testing directly impacts the performance of deep learning models in all environments. A few drops of rain or a snowflake on the sensor's surface can block or distort perception of objects (Figure 2). Snow or ice can block the traffic signs or lane lines, making the object detection task challenging.

HIGH SCHOOL EDITION
Journal of Student Research
Volume 13 Issue 3 (2024)

Figure 2. Sample Image from datasets. The image on the left side is from KITTI dataset showing sunny day road condition. Image on the right hand is from CADC (Canadian Adverse Driving Condition) dataset showing wet snow and obscure traffic condition.

Researchers have publicly provided a significant number of datasets which are crucial to facilitate this research (Table 1). These datasets for AVs have been collected over time through a carefully orchestrated data collection effort that involves equipping vehicles with various sensors and driving them in urban and rural areas. However, most of these commonly available industry standard datasets for AVs are from normal weather conditions and fail to cover harsh weather conditions such as snow, dense fog, or heavy rain. While there are a few datasets gathered in the last 3-4 years from harsh weather conditions, the volume is still very low and some that do exist remain unlabeled. Research and benchmarking on these AV datasets is still primarily geared toward normal weather conditions. Most of the prior research involving adverse weather conditions focused on evaluating one object detection model or specific scenario, such as for foggy conditions or rain/light exposure.

Table 1. List of popular datasets for Autonomous driving.
#1-#5 - Standard Datasets from Normal conditions with minimal weather challenges
#6-#8 - Datasets from Harsh Weather Conditions

	Standard Datasets	**Weather Condition**	**Places / Cities where data were taken**
1	KITTI Vision Benchmark Suit	Normal Weather	City of Karlsruhe, Germany
2	Waymo Open Dataset	Normal Weather	Mountain View, California; Austin, Texas; and Phoenix, Arizona.
3	NuScene	Normal Weather	Boston (Seaport and South Boston) and Singapore
4	ApolloScape Dataset	Normal Weather	Beijing and other three cities in China
5	Lyft Level 5 Open data	Normal Weather	Palo Alto, California, USA
6	Boreas dataset	Harsh Weather	Toronto City, Ontario, Canada

7	Ithaca365 Dataset	Harsh Weather	Ithaca, New York, USA
8	Canadian Adverse Driving Condition (CADC) Dataset	Harsh Weather	Waterloo, Ontario, Canada

In this work, we conducted a more comprehensive evaluation of industry-leading deep learning models to assess the effectiveness of object detection tasks under harsh weather conditions. We performed our research on two models to test our hypotheses because models have different architecture with varying degrees of accuracy, speed, and robustness. Performing research on two models eliminates any bias associated with one set of architecture. We selected the Faster R-CNN and the SSD models for the evaluation (Table 2). We considered the Faster R-CNN for its precise object localization, versatility, and widespread adoption. The SSD was considered for its real-time capabilities and balance between speed and accuracy. These two models differ in the backbone CNN model used and the overall architecture for object detection. The CNN backbone is pretrained on standard COCO datasets, providing the overall model with a starting point for object detection. The Faster R-CNN model uses ResNet and the SSD model has VGG16 as backbone. The SSD has 'one stage' architecture while the Faster R-CNN has 'two stage' architecture for object detection. In 'one stage' architecture, the classification of the objects and sizing of the bounding boxes are carried out in one step. In 'two stage' architecture, the identification of the region of interest occurs first, i.e., if there is an object or just background. In the second stage, the identified region of interest is classified as an object and a bounding box is created.

Table 2. Object detection deep learning models used in the experiment. The architecture, backbone CNN, parameters in Million as well as benchmark mAP (Mean Average Precision) on COCO dataset.

Model Name	Architecture	Backbone	Parameters	Coco mAP
Faster RCNN	Two-stage	ResNet50	43.7	46.7
SSD	One-stage	Vgg16	35.6	25.1

We used two datasets for this experiment. The first dataset from KITTI was from normal weather conditions and it represents the industry standard used to train and evaluate object detection models in AVs. In the first steps of the experiment, we trained and evaluated the three models on KITTI datasets to establish benchmark accuracy. The second dataset was from harsh weather conditions for which we used the Canadian Adverse Driving Condition (CADC) dataset. The annotation data format was not compatible with the training platform used, necessitating the manual annotation of 370 images. In the second step, we evaluated the models from step 1 with CADC dataset to test the first part of our hypothesis that models trained on standard datasets are not effective in detecting objects in harsh weather. In the third and final step, we trained and evaluated the two models on the harsh weather dataset to assess whether object detection accuracy improves when these models are trained with a relevant dataset. This step was essentially done to test and confirm the second part of our hypothesis. The evaluation of object detection models with CADC dataset suggests that models are not effective in detecting traffic objects in harsh weather conditions. However, after training the same models with CADC dataset, their detection capabilities significantly improved.

Volume 13 Issue 3 (2024)

Methods

We conducted an experiment to evaluate the effectiveness of industry-standard object detection models in the detection of traffic objects, such as cars and pedestrians, in harsh weather conditions (Figure 3). We divided the experiment into three distinct scenarios to clearly test our hypotheses. In Scenario 1, we carried out training and evaluation of object detection models with 2D KITTI dataset to establish the benchmark for accuracy with normal weather conditions. In Scenario 2, we evaluated the trained models from scenario 1 with CADC dataset to test our first hypothesis. Finally, in Scenario 3, we finetuned the models with CADC dataset and then evaluated the impact of training with harsh weather dataset on object detection. This is to test our second hypothesis. We utilized the Roboflow platform to preprocess the datasets and the PyTorch platform to obtain object detection deep learning models for our training and evaluation steps. To visualize the training and evaluation results, we employed a Google Colab notebook, where all the necessary Python code was compiled.

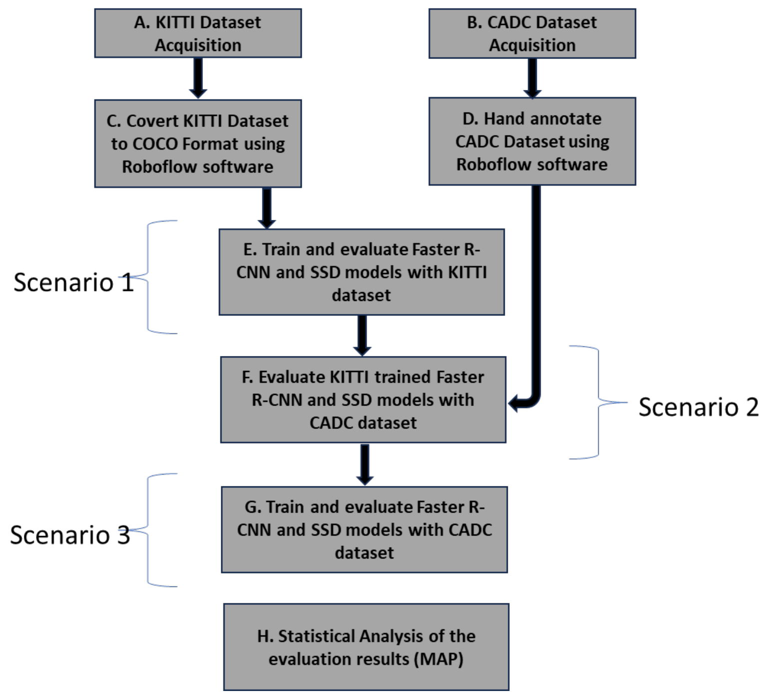

Figure 3. High level flow to outline the steps involved in this research. Step A and B are for downloading the dataset from KITTI and CADC. Step C and D are to pre-process the data for PyTorch platform using Roboflow software. Step E, F and G are training and evaluation steps for each of the three scenarios. Step H is for calculating confidence interval for the mAP results.

Data Acquisition and Preprocessing

We employed 2D images from the KITTI dataset in this experiment for Scenario 1 to train and establish a baseline benchmark for the Faster R-CNN and the SSD models. 7481 images available on the KITTI website were not in a format compatible with common object detection platforms like PyTorch. Therefore, we utilized a more compatible Pascal VOC format which is available at Kaggle. This dataset consists of images in png file format and annotations (bounding boxes and classes) in Pascal VOC XML formats. Therefore, we further transformed this dataset in the Roboflow platform to convert the images from png to jpg file format along with the creation of annotations file in COCO format. KITTI data comprises 7,481 images with 51,868 annotated objects spanning 9 classes.

We downloaded the CADC Dataset in three separate folders, capturing images from March 6th, 2018; March 7th, 2018; and February 27th, 2019. The dataset includes images from 8 cameras, 1 LIDAR, and 1 IMU. To make the dataset compatible with PyTorch, we utilized the Roboflow platform for conversion. While there were thousands of images in the downloaded Waterloo dataset, many of them depicted very similar scenes. Therefore, we filtered out about 10 – 15 unique images per scene for training and testing purposes. Subsequently, we uploaded these images to Roboflow, where we hand annotated all the images. This process was time-consuming as we had to manually place a bounding box around every object that we were detecting in the algorithms. The bounding boxes are needed to precisely fit the object for optimal training and testing. In total, this manual annotation process resulted in almost 2500 annotations across all 1108 images. Both KITTI and CADC datasets were split to create unique sets of training (70%), validation (20%) and testing (10%) data to be utilized during training and evaluation steps.

Model Selection and Transfer Learning

We utilized PyTorch, a leading platform in deep learning research, for our experiment, given that most of the industry-leading object detection models are available at this platform. The two models chosen for evaluation on harsh weather data were pre-trained using the COCO dataset, allowing us to leverage transfer learning. Transfer learning is a widely used concept in machine learning where a model trained on one task is reused or adopted for another related task. The advantage lies in the fact that instead of starting the training and learning process from scratch, transfer learning leverages knowledge gained from solving one problem and applied to a different yet related problem. This approach is well suited for this research where we utilize models pre-trained on object detection tasks for a wide class of objects and then further train it with KITTI and CADC dataset to evaluate the performance. The pre-trained Faster R-CNN model has the mAP of 46.7% and the SSD has accuracy of 25.1%.

Model Training and Hyperparameters

After preparing the datasets and selecting pre-trained models, we proceeded to train the models to finetune their parameters (weights and biases) to learn detecting traffic objects in KITTI and CADC dataset. During training, data is fed in mini-batches through the model network, computing the loss for each mini-batch and optimizing parameters. The selection of hyperparameters plays a crucial role in the performance of object detection models. These parameters are not learned during training but are set at the start of the training process which controls various aspects of the learning process. We maintained a similar setting for hyperparameters during both training and evaluation steps to ensure consistency of results for all iterations. Below are the key hyperparameters used for training:

- Batch size refers to the number of images passed in each forward and backward pass of training step, which updates the model weights. It influences the speed and memory requirement of training. We opted for a batch size of 5, which minimized memory requirement but extended the training duration.
- The number of epochs defines how often the entire training data is processed during training. After a few trial iterations of training and plotting the loss curve, we started to see a plateau in the loss curve after 18-20 epochs for the Faster R-CNN model. Thus, we set 20 epochs for all our training iterations with the Faster R-CNN model. Losses during training with the SSD models stabilized around 28-30 so we set 30 epochs for training with the SSD models.
- The optimizer is an algorithm used to minimize the loss function. It updates the network's weights based on the loss gradient. We used the standard Stochastic Gradient Descent (SGD) optimization

algorithm. Learning rate controls how much we adjust the model's parameters in response to the estimated error. It helps optimal performance of the model during training. We also used a learning rate scheduler to decrease the learning rate every 3 epochs.

Training and Evaluation Metrics

Throughout the training of object detection models, losses are assessed and monitored to get valuable insight into the model's performance, guiding the learning process and ultimately improving the accuracy of object detection. Losses are computed for each batch of training and validation datasets by comparing the model's predictions to the ground truth. Loss curves for both training and validation steps were plotted to track losses as training progressed. Training would be considered complete when losses start to stabilize and show diminishing improvements over several epochs. We also monitored for signs of overfitting, when validation loss may start to get worse, prompting us to stop the training process.

All three scenarios in this experiment had evaluation steps to measure the effectiveness of object detection on a trained model on testing dataset. The AP (Average Precision) stands out as the most widely utilized metrics for evaluating detection accuracy across various annotated datasets employed in object detection benchmarking and research. A crucial indicator of a deep learning model's effectiveness is its precision level, quantifying the ratio of accurate predictions to the total predictions made. Recall, on the other hand, refers to a model's ability to identify all relevant instances, representing the rate at which a model makes accurate positive predictions relative to the number of known facts. For comparing the performance of deep learning models, researchers use the mean average precision (mAP), which is the average of precision scores across all classes. This serves as a singular metric for final evaluation. We selected mAP as a singular metric to evaluate the two object detection models in harsh weather conditions. The evaluation results were recorded for the three scenarios for each object detection model.

Results

We used the RCNN and the SSD model to calculate mAP to evaluate the performance of object detection models with CADC dataset and use it as the key metric to test our hypothesis. It is important to evaluate models on different sets of unseen harsh weather data to assess how models are generalizing object detection to different datasets as well as to assess any uncertainty associated with the results. We performed 10 iterations of evaluation with different sets of images for each scenario. We used a 'random split' function to generate different sets of training, validation, and testing data to be used in each of the 10 iterations of training and evaluation. We also used manual seed to ensure reproducibility and consistency during training for each of the 10 iterations. Once we obtained 10 sets of mAPs for each scenario, we calculated the mean, standard deviation, margin of error, 95% confidence interval (CI) and the p-value.

After the model training and 10 test runs, we achieved a mean average precision of 0.401 and 0.022 in scenario 1 and scenario 2 in the Faster R-CNN model (Table 3 / Figure 4). We notice a similar contrast in the means between scenario 1 (0.200) and scenario 2 (0.030) in the SSD model. Our results support our first hypothesis that object detection accuracy will decline in harsh weather for models trained on data from normal weather conditions with a p-value of 6.43828E-28 for the Faster R-CNN model and 2.71074E-27 for the SSD model, both significantly smaller than 0.05. In addition, we notice that when the model was trained with harsh weather data (scenario 3), the mean average precision was 0.304, significantly higher than 0.022 in scenario 2 in the Faster R-CNN model. Similarly, in the SSD model the mean in scenario 3 (0.143) was significantly higher than the mean in scenario 2 (0.030). These results support the second part of our hypothesis that object detection accuracy improves as models are trained with harsh weather dataset with a p-value of 9.52453E-21 in the Faster

R-CNN model and 3.33012E-25 in the SSD model, both significantly smaller than 0.05. The mAP is lower in the SSD model than the Faster R-CNN model, but results from both fully support our hypotheses.

Table 3. Mean Average Precision (mAP) of the 3 datasets / scenarios over 10 test runs of different image sets on the Faster R-CNN and the SSD Models.

	Faster R-CNN model			**SSD model**		
	Scenario 1	**Scenario 2**	**Scenario 3**	**Scenario 1**	**Scenario 2**	**Scenario 3**
Image set 1	0.414	0.023	0.275	0.194	0.029	0.141
Image set 2	0.412	0.022	0.288	0.201	0.031	0.141
Image set 3	0.388	0.022	0.305	0.209	0.035	0.136
Image set 4	0.403	0.024	0.292	0.199	0.028	0.147
Image set 5	0.389	0.023	0.308	0.197	0.029	0.147
Image set 6	0.390	0.025	0.302	0.196	0.031	0.142
Image set 7	0.405	0.022	0.346	0.198	0.032	0.145
Image set 8	0.410	0.020	0.310	0.205	0.034	0.146
Image set 9	0.395	0.023	0.305	0.205	0.028	0.142
Image set 10	0.400	0.019	0.307	0.202	0.029	0.145
Mean	0.401	0.022	0.304	0.201	0.030	0.143
Standard Deviation	0.0098	0.0018	0.0185	0.0046	0.0024	0.0035
Margin of error for 95% CI	0.0061	0.0011	0.0115	0.0029	0.0015	0.0022
95% Confidence Interval	0.395-0.410	0.021-0.024	0.292-0.322	0.198-0.205	0.029-0.033	0.141-0.147

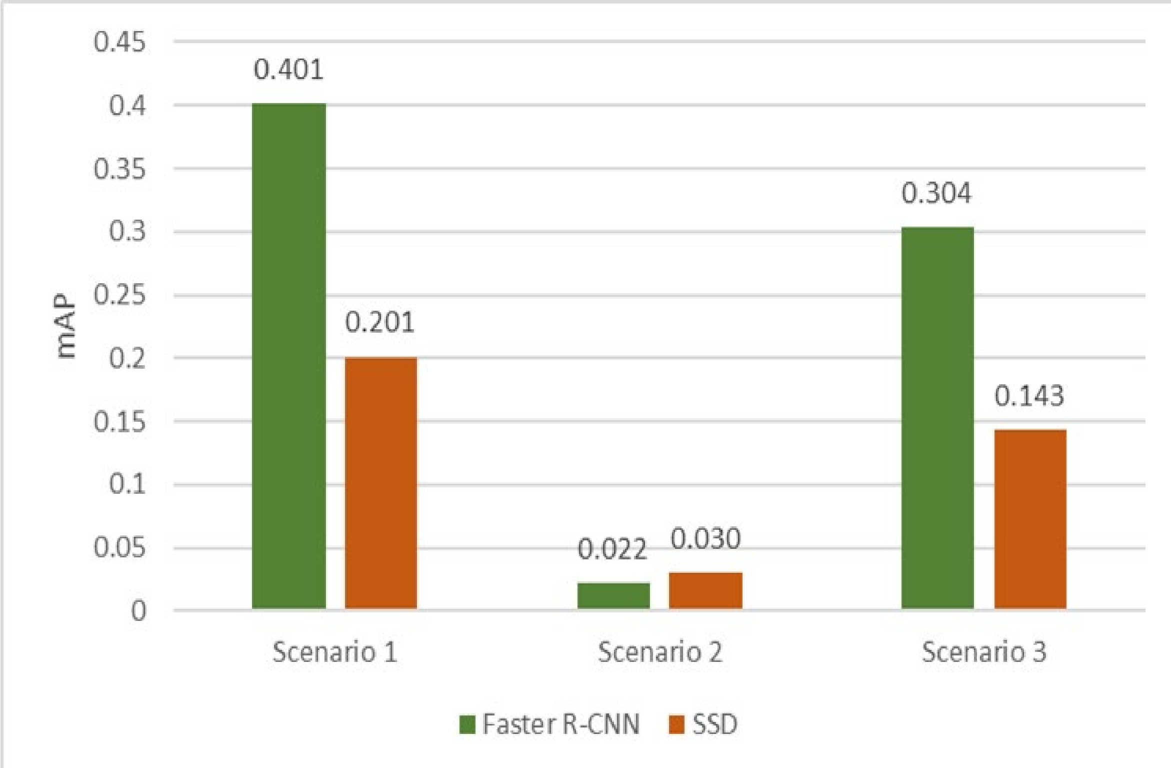

Figure 4. mAP calculated in three scenarios with the Faster R-CNN and the SSD model.

Losses were measured and monitored throughout the training of models with training and validation datasets (Figure 4). For the Faster R-CNN model, losses began to stabilize around 20 epochs with no further improvement for either KITTI and CADC datasets. The SSD model, on the other hand, took 30 epochs to reach stability. Across all the training cycles, we observed that validation losses followed very close to the training losses indicating good generalization of object detection capabilities. This also indicates that models completed their training and the mAP calculated on testing dataset reflects accurate object detection capabilities of the model in this experimental condition. We observed that losses were lower as well as the gap between training and validation losses were narrower with the Faster R-CNN compared to the SSD model. This explains better accuracy results with the Faster R-CNN model.

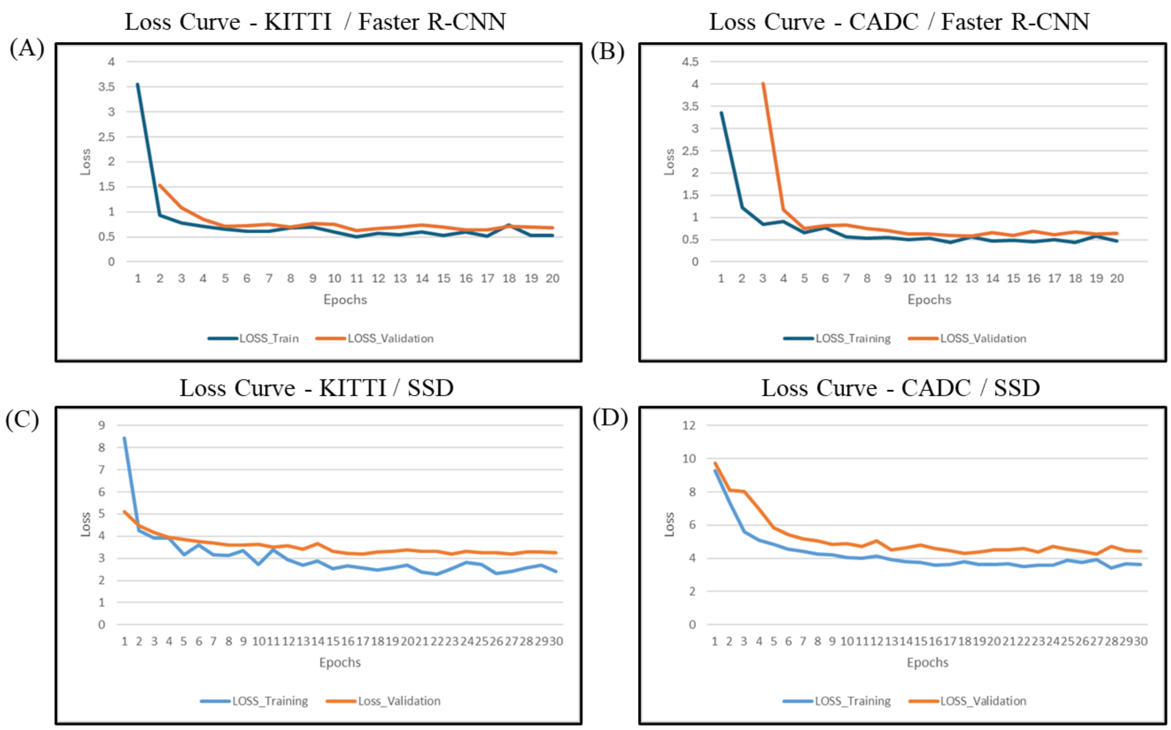

Figure 5. Training and validation losses measured during training.
4A: Losses measured during training of the Faster R-CNN model with KITTI Dataset over 20 epochs.
4B: Losses measured during training of the Faster R-CNN model with CADC dataset over 20 epochs.
4C: Losses measured during training of the SSD model with KITTI dataset 30 epochs.
4D: Losses measured during training of the SSD model with CADC dataset 30 epochs.

Discussion

Both the Faster R-CNN and SSD models had an extremely low mAP in Scenario 2, where we performed object detection evaluation on the harsh weather CADC dataset using models trained with KITTI data. This supports our first hypothesis that leading object detection models, primarily trained on datasets reflecting normal weather conditions, prove ineffective in detecting objects in harsh weather conditions. The CADC dataset is from harsh Canadian winter weather conditions, and we observed with many images that snowflakes are melting into ice-slurry on the camera's optical window and form an opaque blockage on the traffic objects. This can deteriorate image intensity and obscure the edges of the pattern of certain objects which leads to detection failure.

The mAP showed a significant improvement in Scenario 3 for both the models, where we trained the model with CADC data and then performed evaluation. This improvement underscores the fundamental idea of CNN based deep learning models where it learns hierarchical representation of the features from the input data. By exposing the model to a diverse set of training data, the deep learning model learns to generalize its understanding of the underlying patterns which develops the ability of the model to make accurate predictions on new, unseen data.

The faster R-CNN model performed better than the SSD model during training with lower loss values as well as during evaluation with higher AP. This can be attributed to the two-stage architecture of the Faster R-CNN model as compared to the relatively simpler architecture of the SSD model where it predicts the bounding boxes and classification in a single pass. The one-stage architecture makes the SSD model faster, but it comes at the expense of accuracy.

The training and evaluation results from the KITTI dataset were the best, primarily driven by the dataset's volume and number of annotated objects. We had 7,481 images with approximately 52K annotated objects in the KITTI dataset, compared to 1108 images with 2.5K hand annotated objects within the CADC dataset. The discrepancy indicates that object detection precision on harsh weather dataset will improve with the increase of training datasets. The limited number of frames within the dataset for harsh weather used in this work is due to the highly tedious nature of hand-annotating images. Our recommendation is to develop improved systems for labeling data and the capability to format labeled data to be compatible with deep learning platforms. The CADC dataset is labeled by Scale AI, but the format is not compatible with the PyTorch deep learning platform. Consequently, we had to undertake manual annotation and formatting, requiring multiple days of effort.

It is essential to have enough data covering each type of weather conditions (heavy rain, snow, fog, sandstorm, darkness, bright sunlight etc.) and test almost every algorithm on these datasets to expedite the research of AVs in harsh weather conditions. The majority of the available harsh weather datasets are currently collected by researchers in their nearby area by driving a vehicle equipped with cameras, LIDARs, and other sensors. Another increasingly popular method of data acquisition is through simulation platforms like Carla and AirSim, where researchers can create custom-designed, complex road environments with adjustable weather conditions. In these simulation platforms, a variety of vehicles are available to drive, equipped with a choice of sensors to collect data for subsequent use in research focused on perception tasks.

Conclusion

This experiment substantiates that object detection capabilities developed with data from normal weather conditions are not sufficient for harsh driving conditions. It also confirms that deep neural networks have the capability to detect objects in harsh driving conditions if trained with relevant data. It is encouraging to see the efforts that researchers and AV manufacturers have been putting into gathering the harsh weather data and its subsequent annotation, but we still have a long way to go.

Acknowledgments

Noah Curran, PhD from the University of Michigan, USA, has been my mentor throughout this project, offering invaluable guidance on research methodology, referencing relevant research papers, and conducting the final review of the research paper.

References

AV TEST Initiative | Automated Vehicle Tracking Tool | NHTSA. (n.d.). Retrieved January 29, 2024, from https://www.nhtsa.gov/automated-vehicle-test-tracking-tool

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., & Koltun, V. (2017). *CARLA: An Open Urban Driving Simulator* (arXiv:1711.03938). arXiv. http://arxiv.org/abs/1711.03938

Elmassik, Z., Sabry, M., & El Mougy, A. (2022). Understanding the Scene: Identifying the Proper Sensor Mix in Different Weather Conditions: *Proceedings of the 14th International Conference on Agents and Artificial Intelligence*, 785–792. https://doi.org/10.5220/0010911900003116

Fursa, I., Fandi, E., Musat, V., Culley, J., Gil, E., Teeti, I., Bilous, L., Sluis, I. V., Rast, A., & Bradley, A. (2021). *Worsening Perception: Real-time Degradation of Autonomous Vehicle Perception Performance for Simulation of Adverse Weather Conditions* (arXiv:2103.02760). arXiv. http://arxiv.org/abs/2103.02760

Geiger, A., Lenz, P., Stiller, C., & Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, *32*(11), 1231–1237. https://doi.org/10.1177/0278364913491297

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). *SSD: Single Shot MultiBox Detector* (Vol. 9905, pp. 21–37). https://doi.org/10.1007/978-3-319-46448-0_2

Padilla, R., Netto, S. L., & Da Silva, E. A. B. (2020). A Survey on Performance Metrics for Object-Detection Algorithms. *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, 237–242. https://doi.org/10.1109/IWSSIP48289.2020.9145130

Pendleton, S., Andersen, H., Du, X., Shen, X., Meghjani, M., Eng, Y., Rus, D., & Ang, M. (2017). Perception, Planning, Control, and Coordination for Autonomous Vehicles. *Machines*, *5*(1), 6. https://doi.org/10.3390/machines5010006

Pitropov, M., Garcia, D., Rebello, J., Smart, M., Wang, C., Czarnecki, K., & Waslander, S. (2021). Canadian Adverse Driving Conditions Dataset. *The International Journal of Robotics Research*, *40*(4–5), 681–690. https://doi.org/10.1177/0278364920979368

Ren, S., He, K., Girshick, R., & Sun, J. (2016). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks* (arXiv:1506.01497). arXiv. http://arxiv.org/abs/1506.01497

Shah, S., Dey, D., Lovett, C., & Kapoor, A. (2017). *AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles* (arXiv:1705.05065). arXiv. http://arxiv.org/abs/1705.05065

Shi, E., Gasser, T. M., Seeck, A., & Auerswald, R. (2020). The Principles of Operation Framework: A Comprehensive Classification Concept for Automated Driving Functions. *SAE International Journal of Connected and Automated Vehicles*, *3*(1), 12-03-01–0003. https://doi.org/10.4271/12-03-01-0003

Stanley, B. (n.d.). *Autonomous Vehicle Engineers Need to Solve the Weather Problem.* Retrieved December 26, 2023, from https://www.linkedin.com/pulse/autonomous-vehicle-engineers-need-solve-byron-stanley

Zaidi, S. S. A., Ansari, M. S., Aslam, A., Kanwal, N., Asghar, M., & Lee, B. (2022). A survey of modern deep learning based object detection models. *Digital Signal Processing*, *126*, 103514. https://doi.org/10.1016/j.dsp.2022.103514

Zhang, Y., Carballo, A., Yang, H., & Takeda, K. (2023). Perception and sensing for autonomous vehicles under adverse weather conditions: A survey. *ISPRS Journal of Photogrammetry and Remote Sensing*, *196*, 146–177. https://doi.org/10.1016/j.isprsjprs.2022.12.021

Zhou, X., Wang, D., & Krähenbühl, P. (2019). *Objects as Points* (arXiv:1904.07850). arXiv. http://arxiv.org/abs/1904.07850