CASE STUDY

Improving Object Detection Performance by Leveraging Synthetic Data

Applied Intuition

December 2021

Abstract

An object detection model needs to be trained with significant amounts of labeled data before it can play an effective role in an autonomous system's perception stack. To enable object detection models to reach high levels of performance, the training data needs to be sufficiently diverse and capture long-tail events and rare classes. Collecting this type of data is often difficult, expensive, and time-consuming, as long-tail events and rare classes occur, by definition, less frequently than most other events and classes. Class imbalances are thus a fundamental challenge to training and deploying effective and safe perception systems. Applied's customers are utilizing synthetic data for developing their perception systems by creating labeled data to address imbalances found in real-world datasets.

To demonstrate this training use case, Applied Intuition's perception simulation team has conducted a case study that looks specifically at the issue of class imbalances and how it can be addressed by leveraging synthetic data. The focus is on a class imbalance found in a popular real-world dataset, nuImages by Motional, in which well-represented classes such as cars occur over 170 times more frequently than cyclists [1]. The goal is to improve the cyclist detection performance of a Cascade Mask R-CNN model while retaining or improving object detection performance on other classes [2]. To achieve this goal, synthetic data is generated in Applied's perception simulation tool Spectral and used as a supplemental training resource in addition to the nuImages dataset.

The results of the case study show an improvement in both overall object detection (all classes) and cyclist detection, as well as an improvement in the model's ability to handle difficult cases compared to the baseline. In this case, synthetic data provides an effective complement to real-world data, enabling underrepresented classes to be upsampled in a scalable, cost-effective, and rapid manner.

I. Introduction

The goal of this study is to improve a perception algorithm's object detection performance on cyclists while retaining or improving object detection performance on other classes.

This case study was pursued because class imbalances can lead to safety-critical issues in autonomous vehicle (AV) perception systems. Synthetic data provides a direct way to upsample the underrepresented classes in skewed datasets or fill in long-tail events. The most complex cases that present the greatest safety challenges can be hidden in the last few percent of test case errors. These cases typically have a low base frequency in the real-world datasets used to train perception models. For example, cyclists may represent only 0.3% of a real-world dataset (Figure 1), which directly impacts perception model performance and can cause catastrophic implications in safety-critical situations (e.g., fatal crashes between AVs and cyclists).

There are several approaches to mitigating class imbalances in real-world image data. Algorithmic techniques such as using the focal loss function, data augmentation (e.g., cropping and upsampling), and making updates to model architectures are common for training neural networks on imbalanced data [3]. Collecting more real-world data to address these imbalances is also possible, but it is often expensive, not scalable, and slow.

Another common approach for dealing with classes

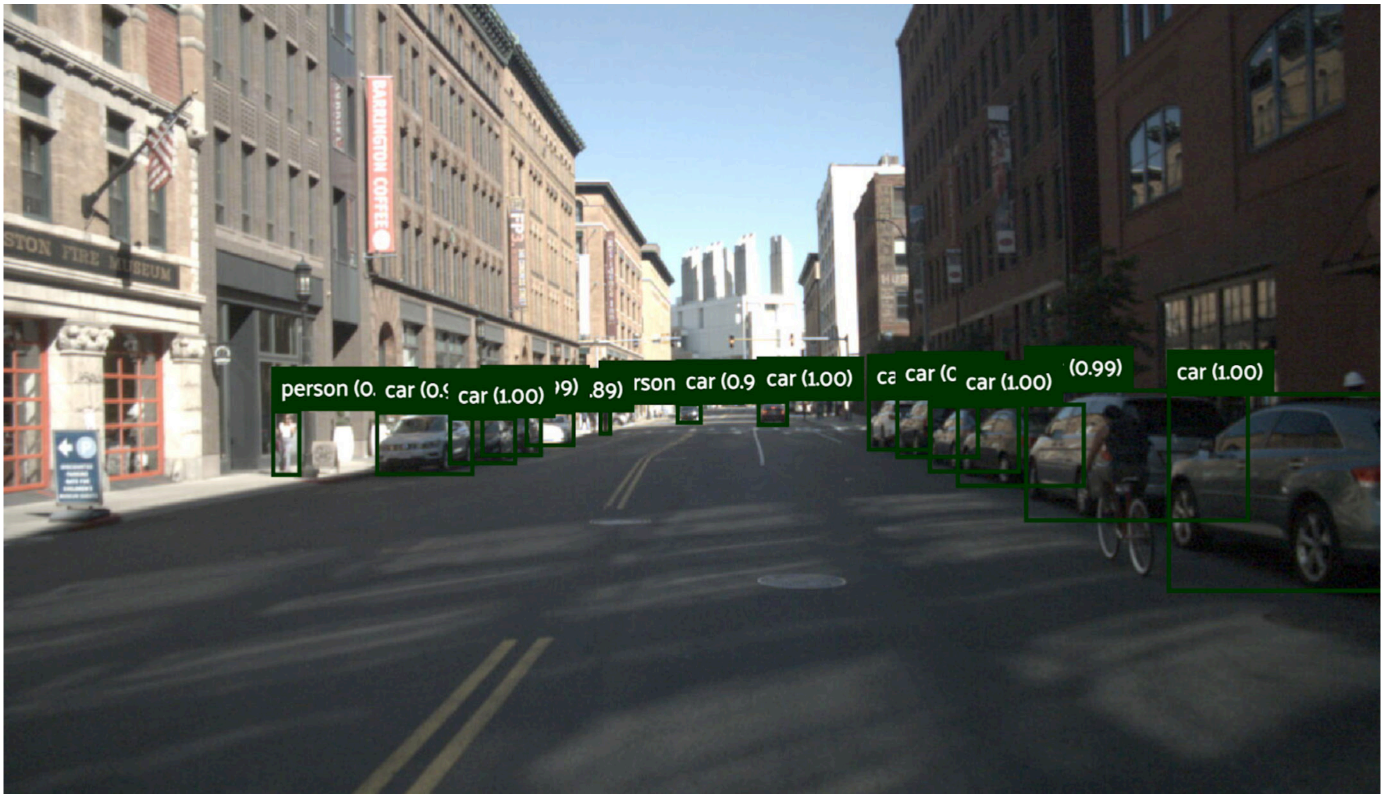


Figure 1: Cyclists are often underrepresented in real-world datasets. This makes it difficult for a perception model trained only on real-world data to detect cyclists.

with little data is using synthetic data as a supplemental training resource in addition to real-world image datasets. This approach has continued to gain popularity over the past few years [5]. There still exists some domain gap between real-world and synthetic data, which is important to acknowledge, but recent methods are overcoming the domain gap with a combination of improved synthetic data and new machine learning training strategies. The simulation-to-real gap (i.e., a degradation in object detection performance due to a difference between the synthetic data used for training and the target domain in the real world) [4] has been well documented and is important to understand in relation to the results presented in the current study.

The state-of-the-art uses of synthetic data are showing that synthetic data can be useful in mitigating class imbalances and addressing areas where real-world data is limited. This study specifically explores whether the use of synthetic data can reduce the amount of real-world data needed to improve a model's object detection performance.

II. Dataset & Methods

This case study uses the nuImages dataset by Motional as a baseline training dataset. nuImages is a real-world dataset of 93,000 two-dimensional annotated images that is commonly used to train and evaluate AV perception algorithms. This study specifically uses the nuImages training set, which consists of 67,279 labeled frames, and the nuImages validation set, which consists of 16,445 images. In the nuImages dataset, the cyclist class is underrepresented compared to other prominent classes. Occurrences of cyclists in the dataset are over 170 times less frequent than other well-represented classes such as cars (Figure 2).

The class imbalance found in nuImages is not unique to this particular dataset. In fact, Motional created the nuImages dataset by actively mining existing data for rare classes such as cyclists. The base frequency with which an AV would encounter cyclists in the real-world is much lower than the base frequency contained in the nuImages dataset.

III. Experimental Setup

The study consists of the following steps:

1. Analyze a baseline model trained only on real-world data from the nuImages dataset.
2. Generate labeled synthetic data that specifically targets the lack of representation of the cyclist class in the real-world dataset. More examples of cyclists are created in the synthetic dataset.
3. Use the above synthetic data as a supplemental training resource in addition to the nuImages data.

In step 3, a Cascade Mask R-CNN model is trained for bounding box object detection and semantic instance segmentation. Results are shown across two types of experiments:

i) Mixed training experiment

- Synthetic data and real-world data are combined into one large training dataset. Batches that contain both real-world and synthetic data are randomly sampled from this dataset during training. The ratio of synthetic to real-world data is adjusted to explore whether using more synthetic data and less real-world data impacts the model's object detection performance.

ii) Fine-tuning experiment

- **Fine-tuning without data ablation:** Models are initially trained to convergence on synthetic data and are then fine-tuned on the real-world data domain. Hence,

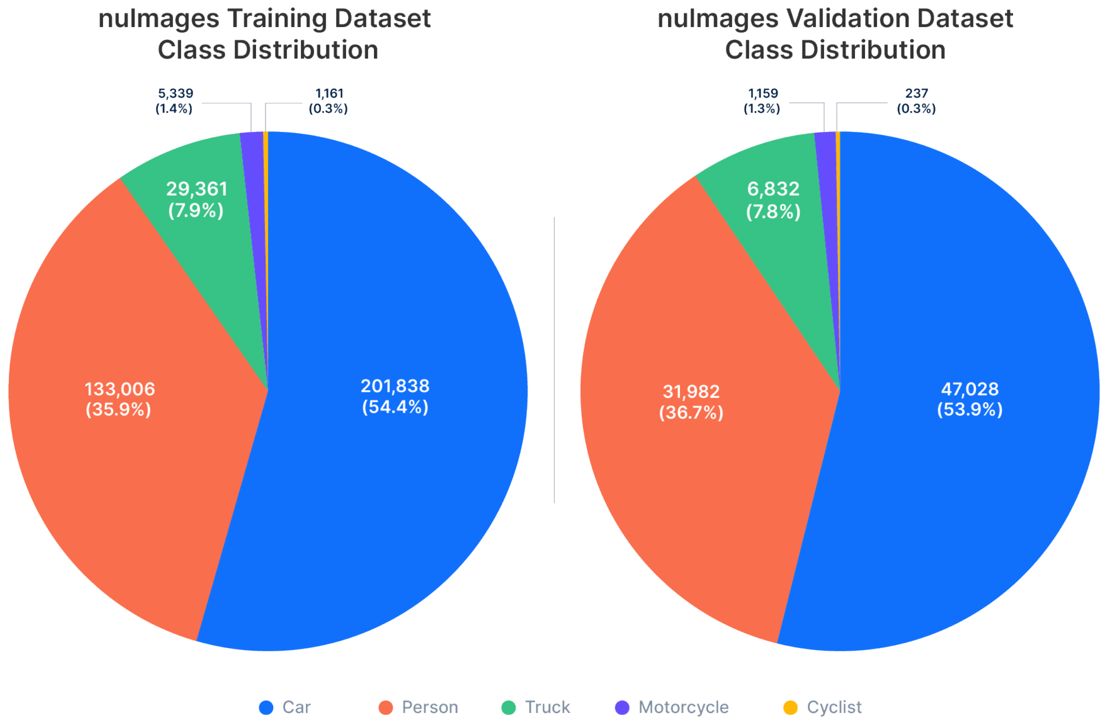


Figure 2: The class distributions for five of the classes contained in the nuImages training and validation sets. People and cars occur more frequently (a total of ~90% of the five classes used in this study). Cyclists occur only 0.3% of the time.

models encounter only synthetic data during pre-training and then encounter only real-world data during fine-tuning. In these trials, the size of the real-world training dataset is not ablated (i.e., downsampled).

- **Fine-tuning with data ablation:** As an extension of the fine-tuning experiment, models are trained on limited amounts of real-world data to explore whether the use of synthetic data diminishes the amount of real-world data needed for training.

For evaluation, bounding box and segmentation mean average precision (mAP) scores (i.e., the measure of the accuracy of object detection) are reported both in aggregate and at a per-class level. All mAP values are reported as averages over an Intersection-over-Union (IoU) value (i.e., the measure of how much the predicted boundary overlaps with the ground truth) ranging from 0.5 to 0.95, with steps of 0.05. All models are trained on the same five classes: cars, motorcycles, trucks, people, and cyclists.

In order to control for independent variables, the following experiment design choices and assumptions are established:

- Model architectures and hyperparameters are held consistent. For the fine-tuning experiment, the learning rate is reduced.
- Models are trained until convergence.
- All models are evaluated on the same evaluation set of real-world data (nuImages-val).

IV. Implementation

1. Baseline model analysis

It first needs to be determined how a perception model reacts to a class imbalance in the real-world nuImages training data. A Cascade Mask R-CNN perception model is trained on this dataset until convergence. The following results are reported (Figure 3).

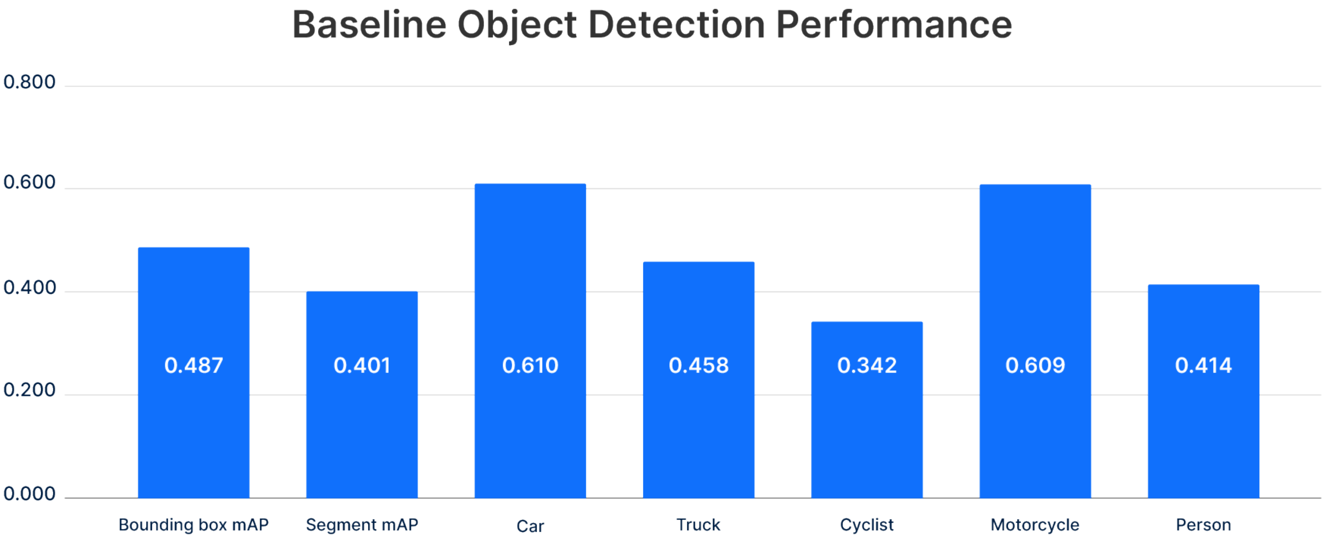
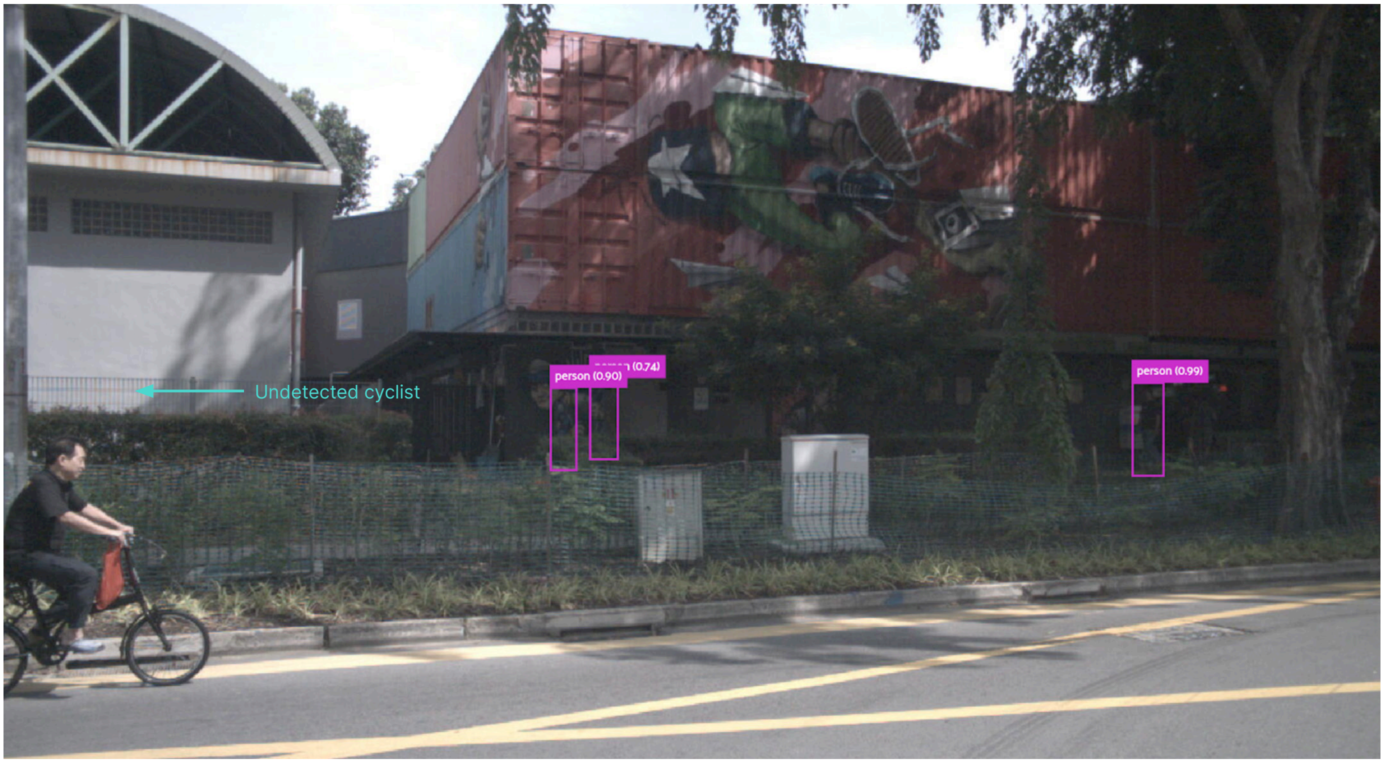
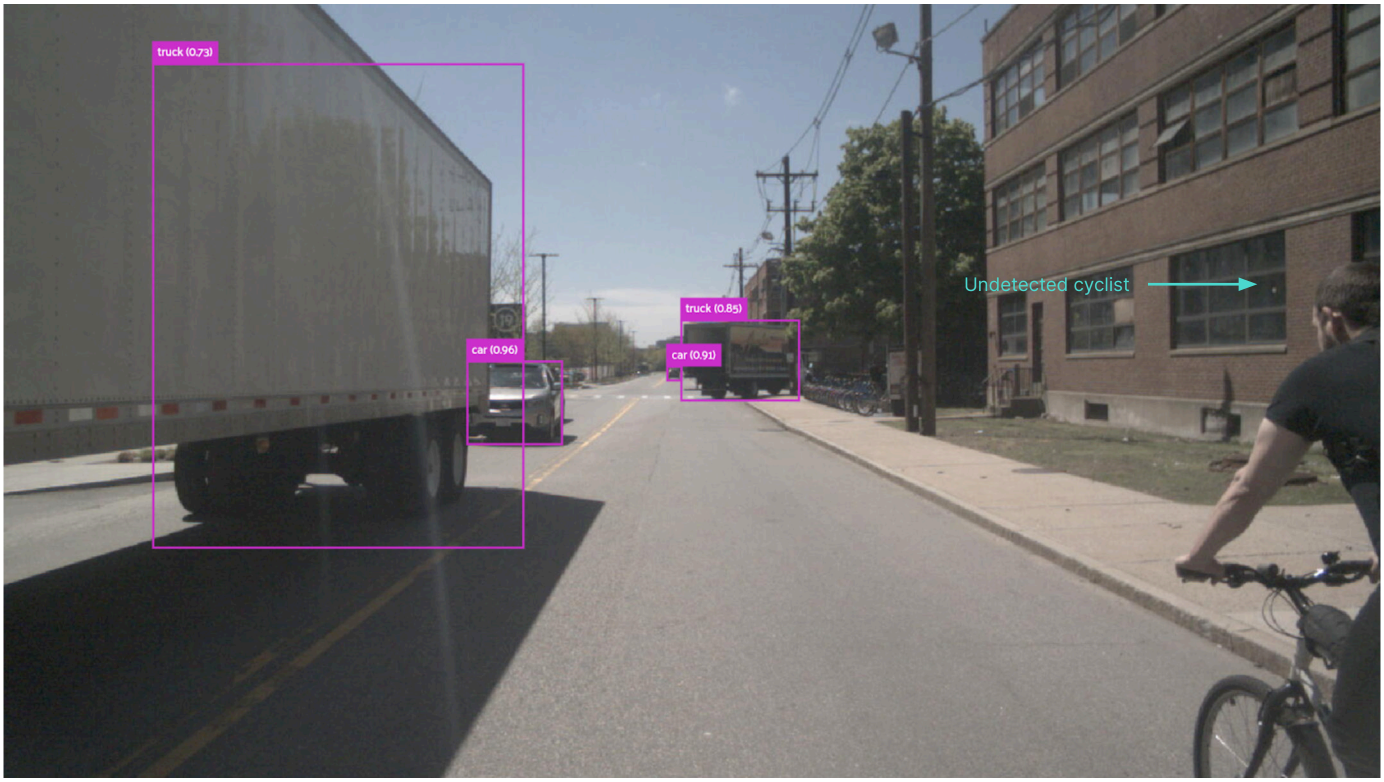


Figure 3: The object detection performance of the baseline perception algorithm when trained on the nuImages training and validation sets. Aggregate performance (bounding box, segment) and per-class performance (car, truck, cyclist, motorcycle, person) are expressed in mAP values as averages over 0:5:0.95 IoU values.


n (0.74)
person (0.90)
person (0.99)
Undetected cyclist


truck (0.73)
car (0.96)
truck (0.85)
car (0.91)
Undetected cyclist

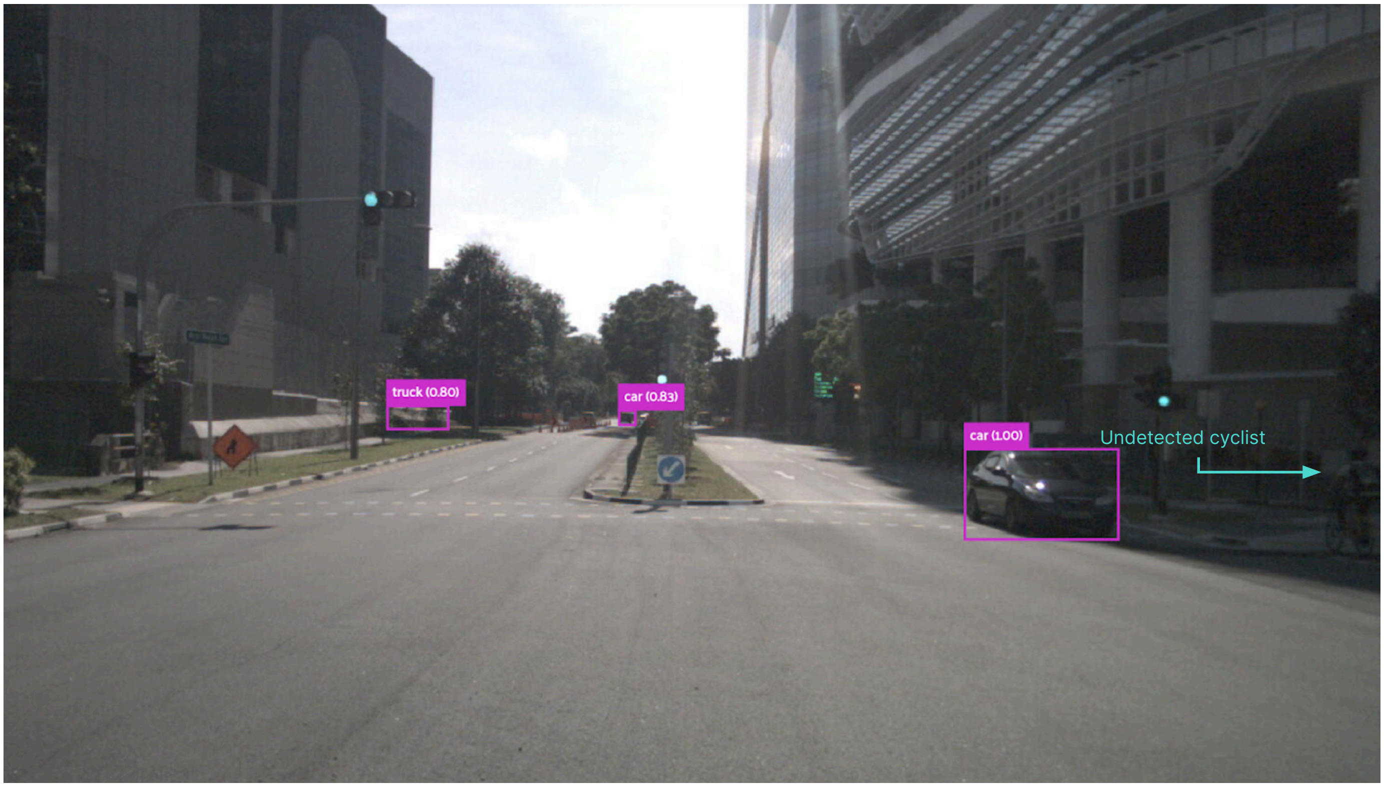


Figure 4: Examples of scenes in the nuImages dataset where cyclists are not detected.

The cyclist class is underperforming quantitatively. When the results are analyzed qualitatively, there are many examples where the perception algorithm misses cyclist detections (Figure 4).

2. Synthetic data generation

Next, synthetic data needs to be generated in order to up-sample the underrepresented cyclist class. The process of synthetic data generation in Spectral can be broken down into three components:

- 3D synthetic environment creation
- Synthetic scenario generation
- Synthetic data generation

i) 3D synthetic environment creation

Generating synthetic 3D environments at scale is challenging because environments need to be representative of the operational design domain (ODD) and be visually rich (i.e., contain variations) for them to be useful for training. The manual creation of the environments is expensive, time-consuming, and especially difficult when trying to create visually diverse worlds in many global ODDs.

Procedural generation enables the generation of large, visually diverse environments within a short period of time. A virtual environment similar to the Boston and Singapore nuImages ODDs is created in order to minimize the effect of an environmental domain gap.

ii) Synthetic scenario generation

To build synthetic training sets, diverse scenes with varying actors (vehicles, cyclists, obstacles), weather conditions, times of day, ego poses, etc., need to be generated. This process is time- and resource-intensive when done manually. At the same time, it is crucial to maintain control of the relative distribution of actors and other scenario parameters and to preserve realism.

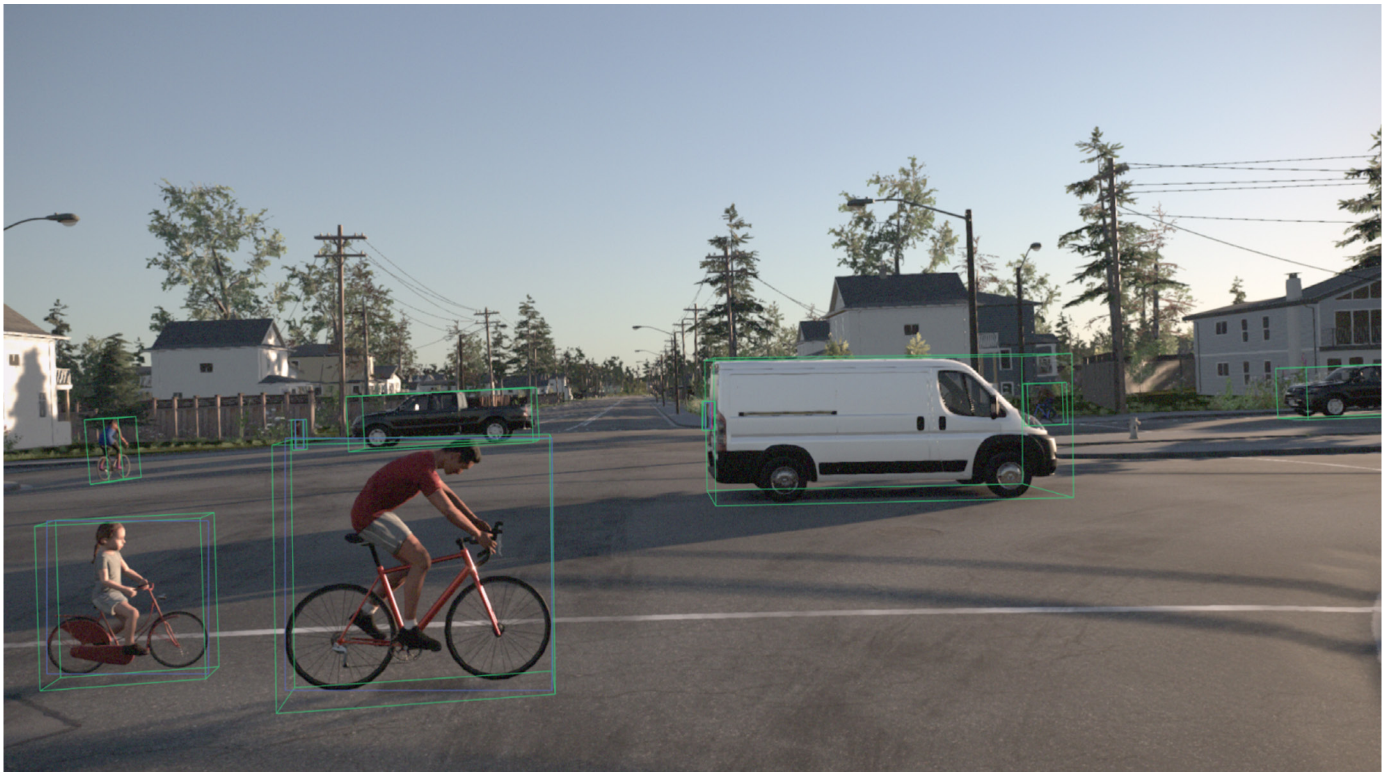


Figure 5 a): Spectral synthetic images using sequential scenarios with per-actor definition.

Automatic scenario generation enables the creation of realistic distributions for all scenario parameters and the sampling from these distributions to achieve coverage of various scenarios. In the synthetic dataset used in this study, three different forms of "scenario" generation were used:

1. Sequential scenario creation by defining individual actor behaviors
2. Sequential scenario creation using traffic generators
3. Non-sequential data frames using distributions and smart actors

Examples of each of these methods are shown in the following images, along with their ground truth data (Figure 5 a) - 5 c)).

Using each of the three methods above, Spectral datasets can be tailored to very specific events, or randomized to produce extremely large-scale data. For this study, a large number of realistic, sufficiently diverse urban traffic scenarios are quickly generated.

iii) Synthetic data generation

Lastly, it can be challenging to generate large amounts of synthetic data at a low cost with fast turnaround times. A synthetic data generation pipeline enables the consecutive generation of different versions of a full synthetic dataset. A perception model is iteratively trained, training results are evaluated, and the results are used to refine the dataset.

This study's dataset of 98,515 labeled frames specifically upsamples the cyclist class and is generated across a variety of map locations on synthetic urban and suburban maps across a variety of weather and lighting conditions (Figure 6). This generation process takes fewer than four hours with horizontal scaling on the cloud. The resulting quantitative model performance metrics then inform ways to improve the dataset content.

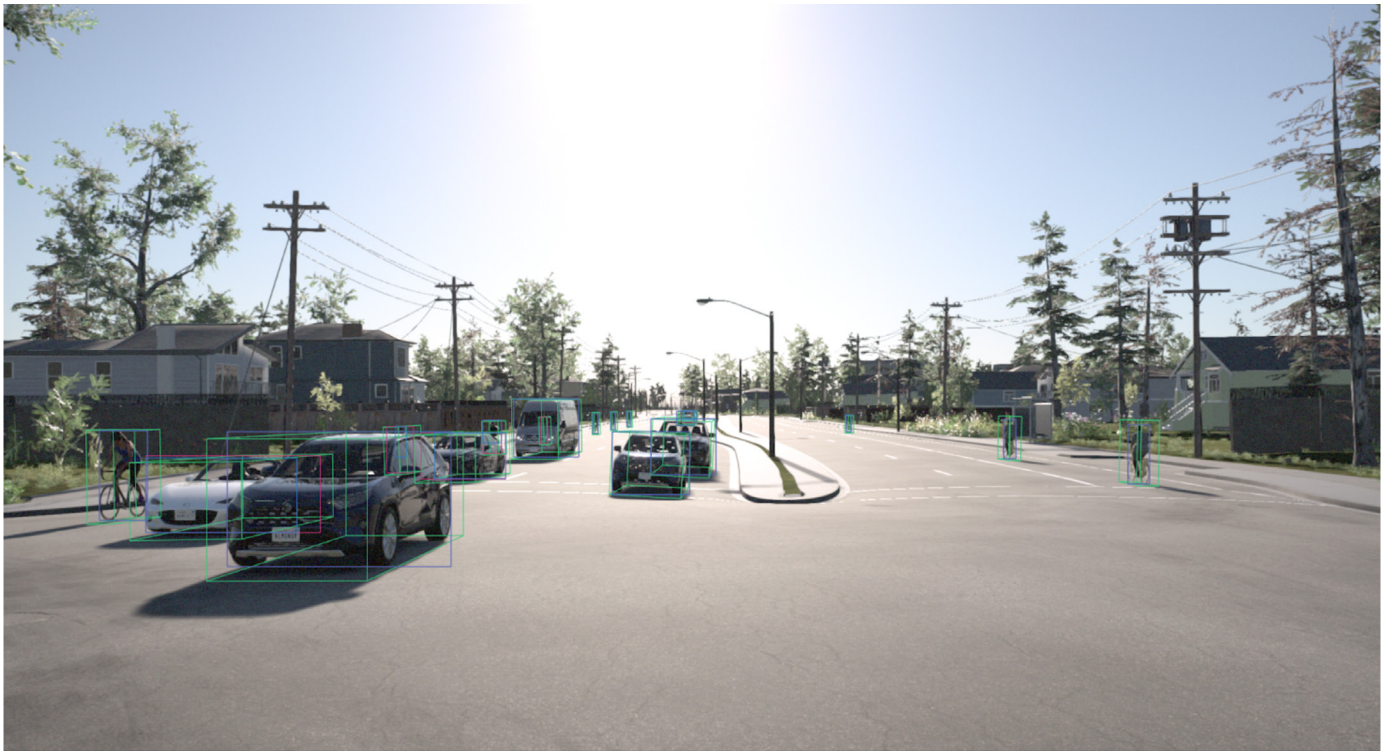


Figure 5 b): Spectral synthetic images using sequential scenarios with per-actor definition.

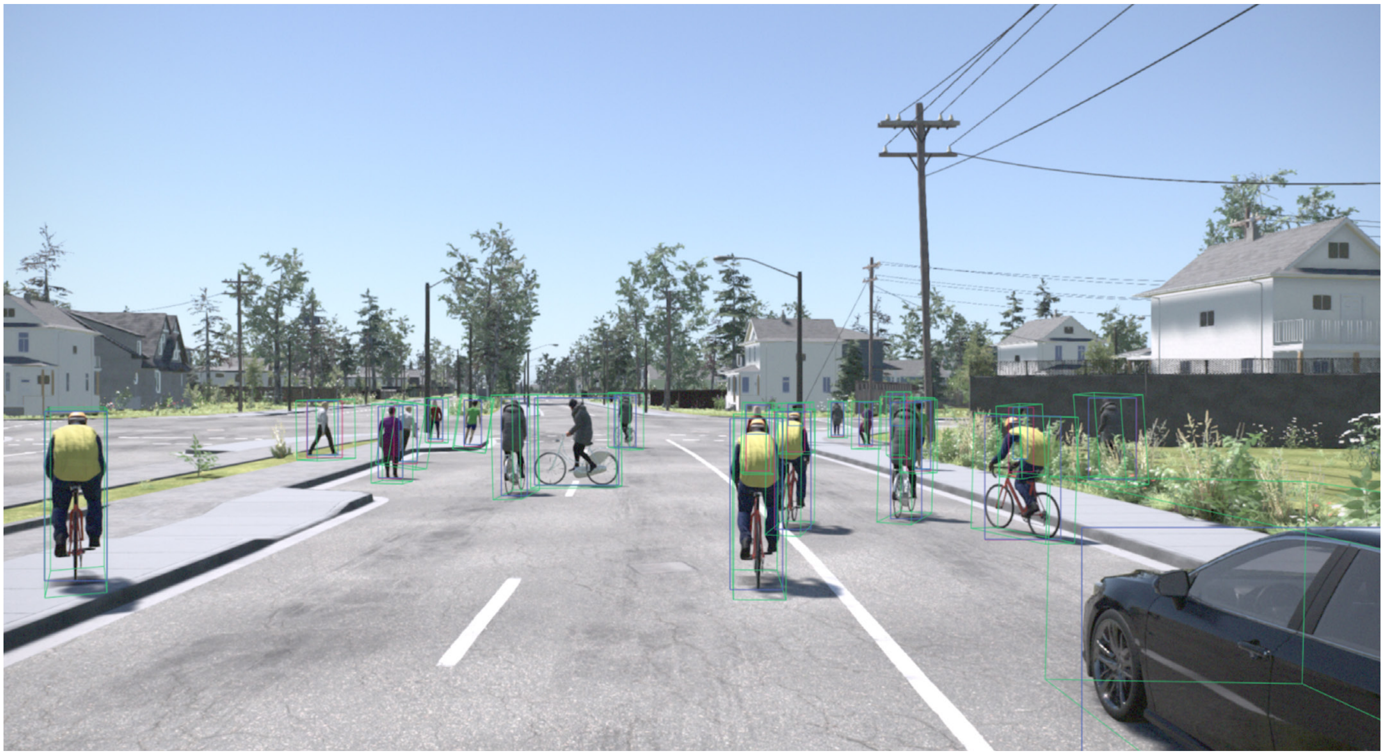


Figure 5 c): Spectral synthetic images using non-sequential frames with randomized smart actors.

3. Perception model training with synthetic and real-world data

The above synthetic dataset is then used to improve model performance in mixed training and fine-tuning experiments. There are two key problems that need to be solved when adding synthetic data to the training set. First, the performance of the model on the class that is being upsampled should increase. Second, the performance across all classes should not degrade due to the addition of synthetic data to the training set.

i) Mixed training experiment

Two trials are conducted and the amount of synthetic data used in each trial is varied. The trials use the following ratios:

- A 0.5:1 ratio of synthetic to real-world data
- A 1:1 ratio of synthetic to real-world data

ii) Fine-tuning experiment

A model is trained to convergence on only the synthetic dataset using a small holdout synthetic set for validation. Without any fine-tuning on real-world data, the model underperforms on the real-world data. This result is expected due to the simulation-to-real gap. Said another way, the model trained on only synthetic data cannot overcome the domain gap to perform well when tested on real-world data.

Three trials are then conducted to fine-tune the model on the following amounts of real-world data:

- Fine-tuning without data ablation: 100% of the nuImages training set
- Fine-tuning with data ablation: 70% of the nuImages training set
- Fine-tuning with data ablation: 50% of the nuImages training set

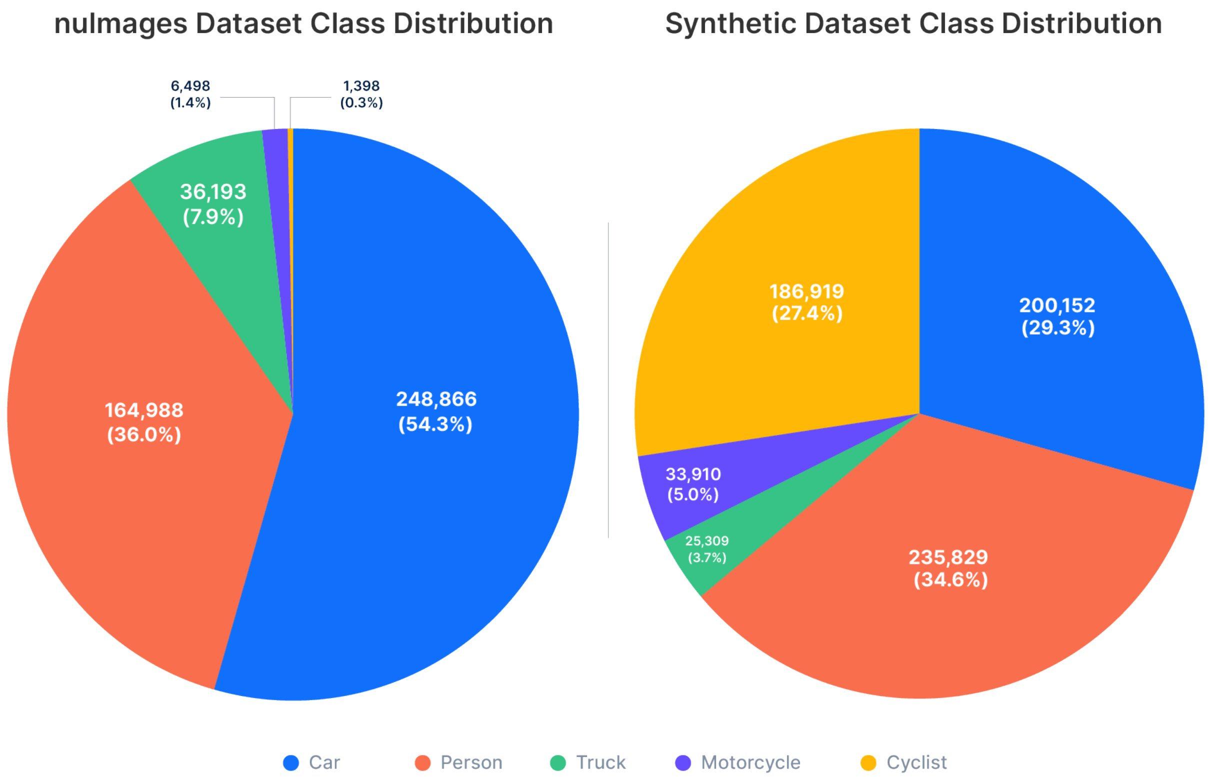


Figure 6: Class distribution of real-world (nuImages) and synthetic datasets. Cyclists (yellow) are upsampled in the synthetic dataset (right).

To examine the results of the fine-tuning trials with data ablation, another baseline model is also trained to convergence on only 50% of the nuImages training set without being pre-trained on synthetic data.

V. Results

This case study demonstrates that pre-training a model on synthetic data and then fine-tuning it to the real-world domain results in more performant models. It also suggests that mixing synthetic and real-world data leads to improvements compared to using only real-world data. Although certain experimental cases may have different best practices, this study's findings can be summarized as follows:

- The overall object detection performance increases when a perception model is trained using either mixed training or fine-tuning methods. The largest and most consistent improvements are seen in the cyclist class (see 1. Quantitative results).
- Object detection model performance is specifically improved on difficult cases when the model is trained using mixed training or fine-tuning methods (see 2. Qualitative results).

1. Quantitative results

i) Overall mAP results

The overall object detection performance of mixed training and fine-tuning experiments is summarized in Figure 7. The best-performing trials of the mixed training and fine-tuning experiments show improvements relative to the baseline model. The fine-tuning experiment without data ablation shows the largest improvement of +0.022 mAP (4.5% improvement). The performance decreases when a higher ratio of synthetic to real-world data is used, potentially due to the domain gap effect.

ii) Results on individual classes

The object detection performance specific to the cyclist class is shown in Figure 8. Both mixed training and fine-tuning experiments show strong performance

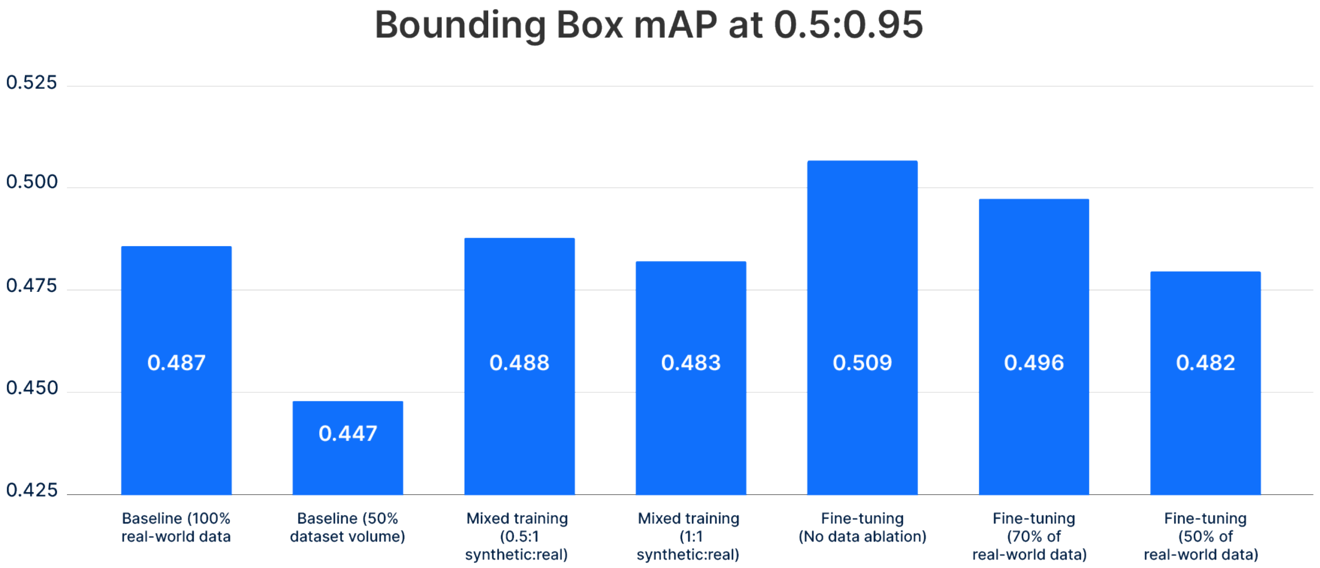


Figure 7: The best-performing trial of the mixed training experiment (0.5:1 synthetic:real) and the two best-performing trials of the fine-tuning experiment (without data ablation and with 70% of real-world data) show improvements relative to the baseline model. The fine-tuning experiment with 50% of the real-world data achieves nearly the same performance as the baseline. It far surpasses the performance of the model trained only on 50% of the real-world data and no synthetic data.

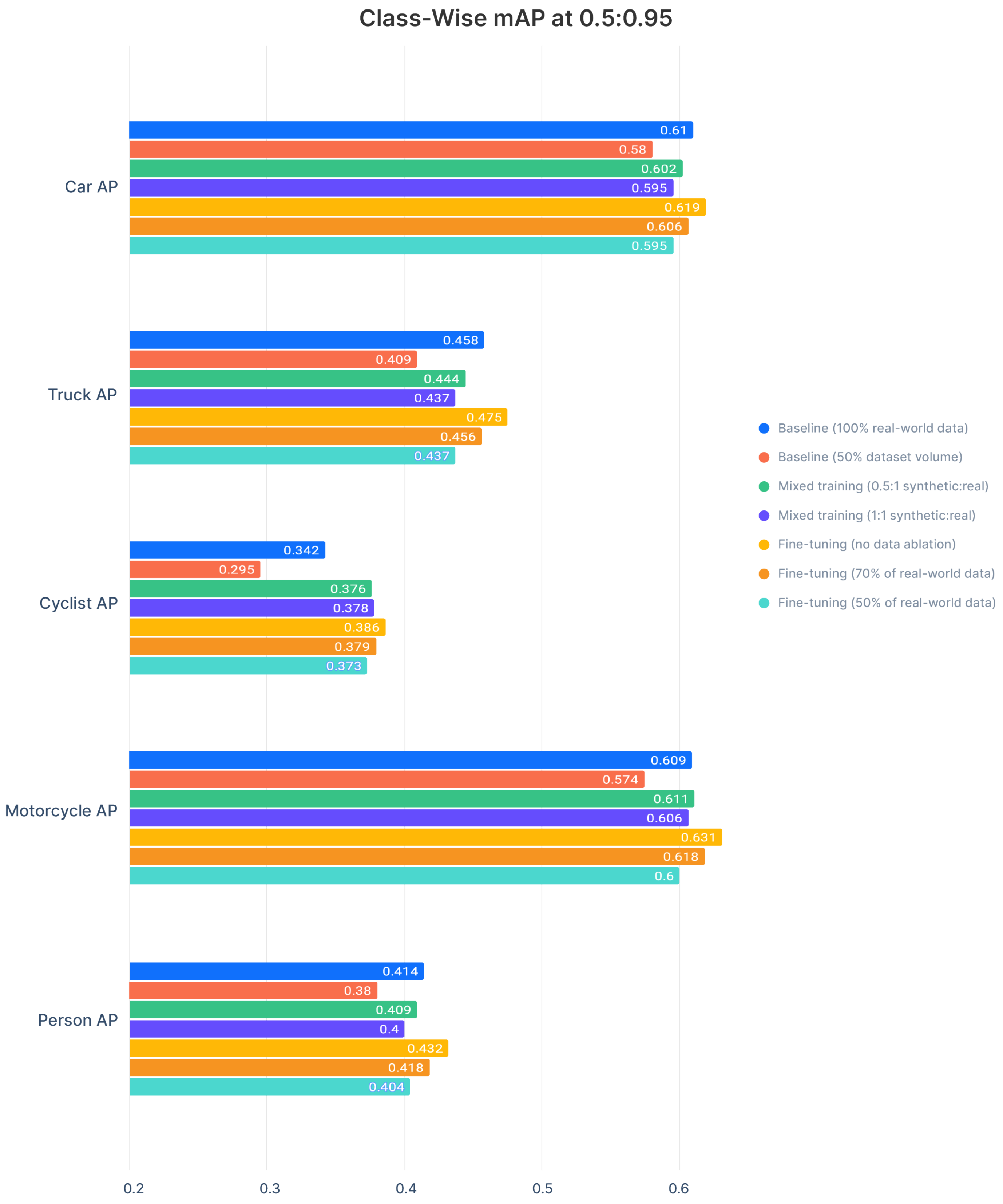


Figure 8: Class-wise mAP scores. Mixed training and fine-tuning experiments improve the mAP scores on cyclists, while improvements are limited on other classes.

improvements on the cyclist class with a mix of limited regressions and improvements on the other classes. The fine-tuning experiment without data ablation shows the highest performance improvement, outperforming the baseline model consistently on all classes.

iii) Effects of data ablation

The two trials of the fine-tuning experiment with data ablation (using 70% and 50% of the nuImages training dataset) outperform and underperform, respectively, in comparison to the baseline model. Specifically, the trial using 70% of the nuImages dataset performs +0.009 mAP better than the baseline overall (Figure 9) and +0.037 mAP better than the baseline on the cyclist class (Figure 8). The trial using only 50% of the real-world data results in a slightly lower performance than the baseline model overall, with a difference of -0.005 mAP (Figure 9). However, the trial still surpasses the baseline performance on the cyclist class of 0.342 mAP by +0.031 mAP (Figure 8).

iv) Mixed training for cyclist class

The mixed training experiment shows detection performance improvements on the cyclist class (Figure 8). The trial with a 0.5:1 ratio of synthetic to real-world data achieves similar overall performance as the baseline model (Figure 7) and improved performance on the cyclist class (Figure 8). The trial with a 1:1 ratio of synthetic to real-world data underperforms when compared to the trial with the 0.5:1 ratio (Figure 7). The larger fraction of synthetic data will at some point degrade the results due to the domain gap. Achieving the right balance of synthetic and real-world data under all cases is an ongoing study.

2. Qualitative results

When examining the differences in object detection performance between models trained with real-world data alone and models trained with supplemental synthetic data, qualitative results suggest that synthetic data can help improve object detection performance in difficult cases. The following images show several qualitative examples from the validation set where cyclists are occluded, at a distance, or in difficult lighting conditions. These qualitative examples show cases in which the baseline model fails to adequately detect a cyclist while the model pre-trained on synthetic data succeeds (Figure 10 a) - 10 e)).

While the quantitative results of this study show

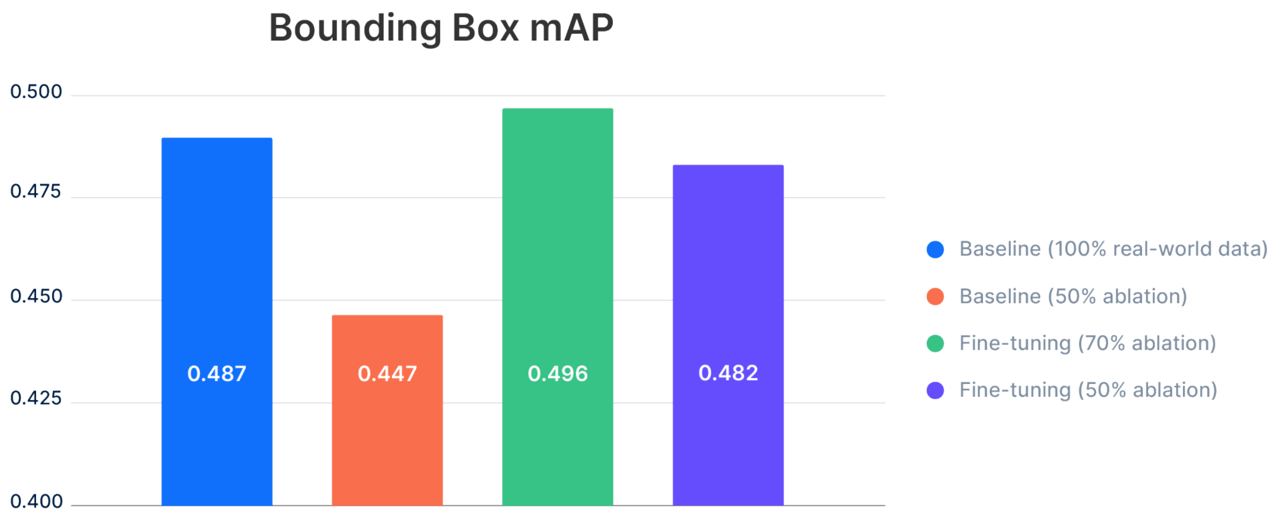


Figure 9: Mean average precision (mAP) scores from the fine-tuning experiment. The mAP score of fine-tuning with 70% of the real-world data (green) outperforms the baseline with 100% of the real-world data (blue).

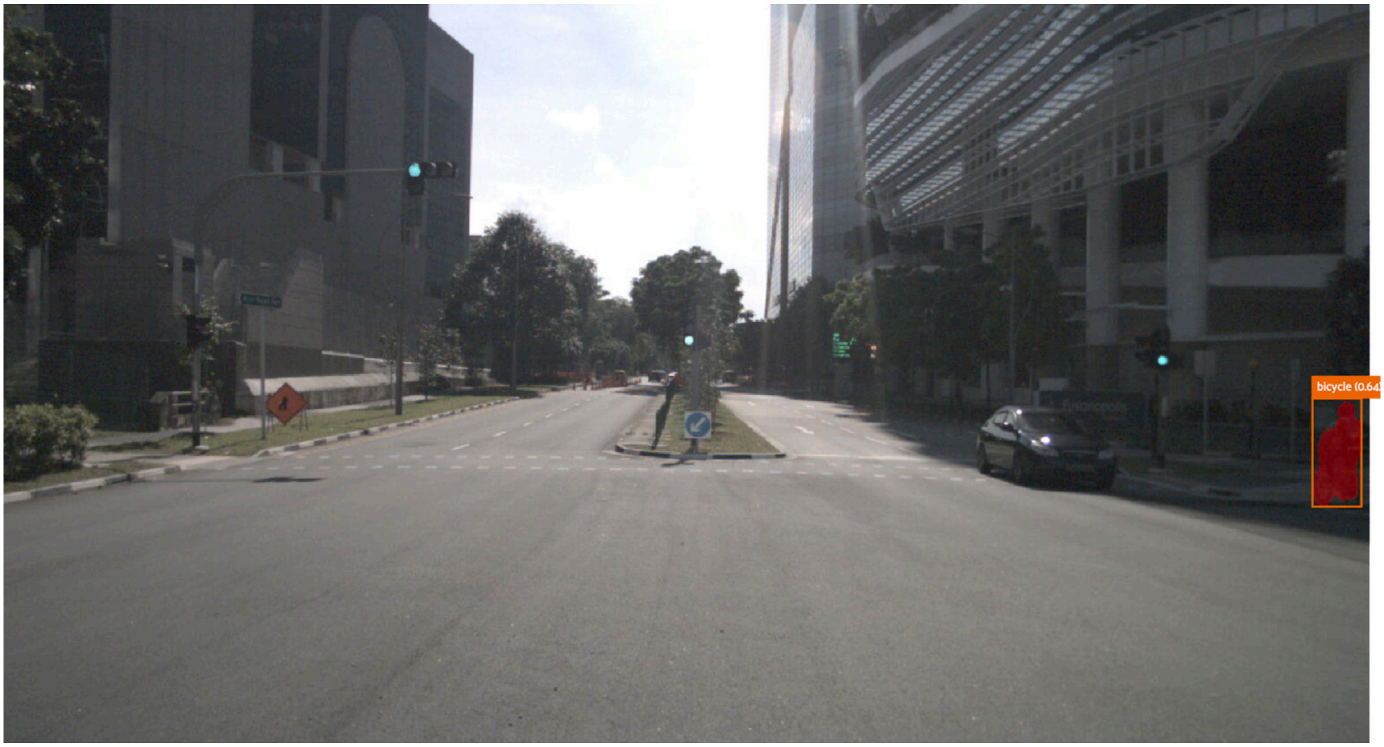


Figure 10 a): The model pre-trained on synthetic data detects a cyclist in the shade..

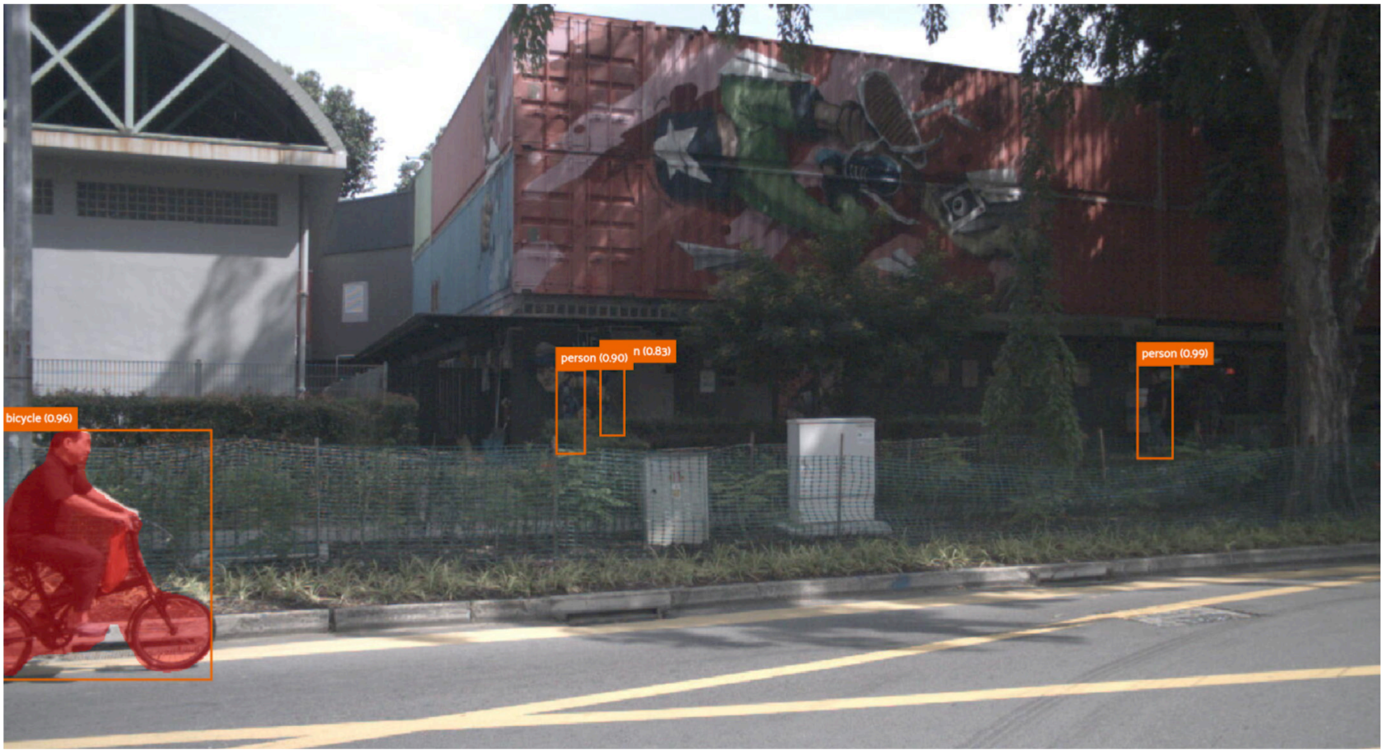


Figure 10 b): The model pre-trained on synthetic data detects a cyclist close to the ego vehicle.

improvements in aggregate, the qualitative results suggest that synthetic data helps improve aw perception model's object detection performance specifically in difficult cases.

VI. Potential Applications

This study shows an early indication that synthetic data can be used in combination with real-world data to train perception models on both nominal and edge cases. Scenes such as fallen objects or live animals in the middle of the road are rare to come by but autonomous vehicles must be prepared to correctly detect and safely navigate around them. Rather than driving hundreds of thousands of miles in the real world to collect a sufficient volume of such long-tail events, synthetic data can be used to create sufficient training datasets that specifically target those cases.

Another advantage of synthetic data is the ability to quickly change environmental conditions such as weather and lighting. This study's qualitative results show that synthetic data can be used to improve cyclist detection performance even in difficult cases such as occlusion and shade. Thus, synthetic data may be used to create a training dataset containing long-tail events of varying environmental parameters to prepare the model for all conditions of the detection.

Finally, certain events are too dangerous to train for (e.g., a pedestrian suddenly jumping onto the road). In these cases, synthetic data becomes the only option for filling out the distribution of cases that an AV perception system is required to be tested against.

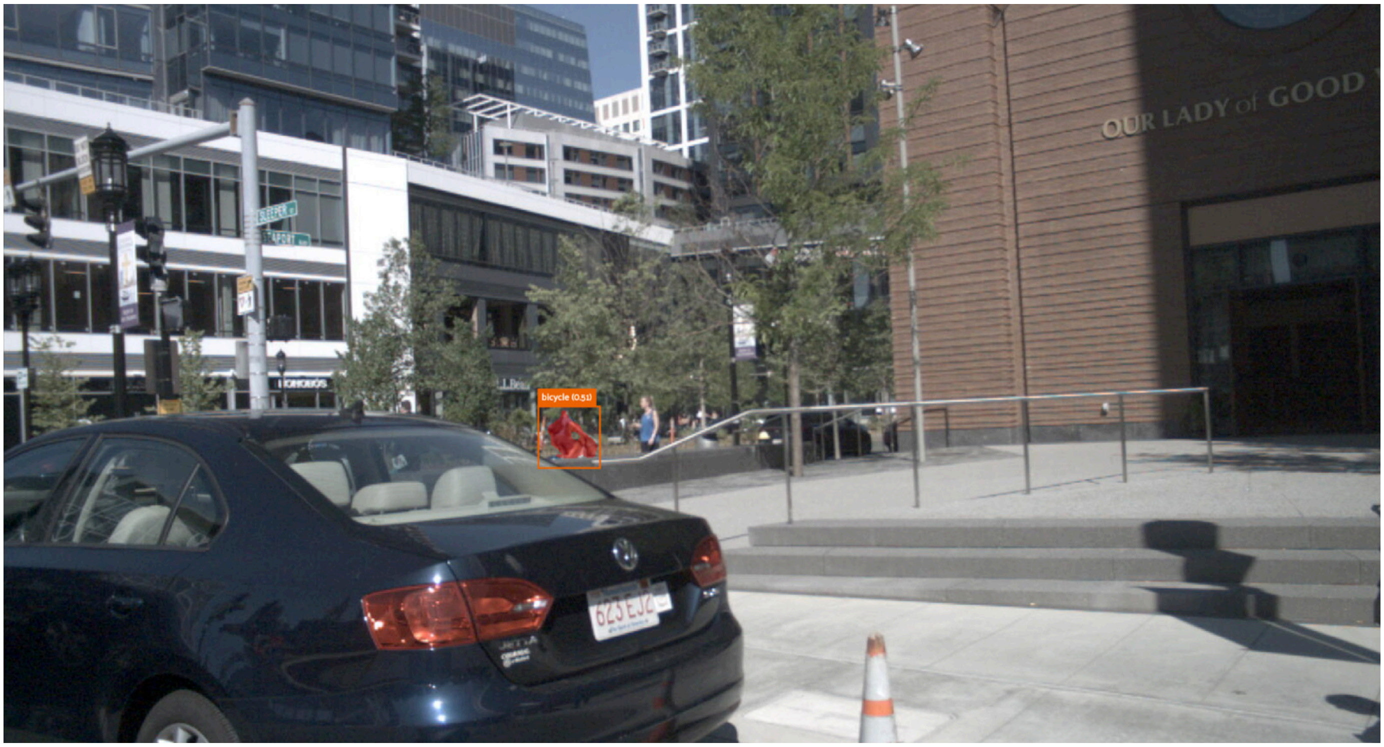


Figure 10 c): The model pre-trained on synthetic data detects a partially occluded cyclist.

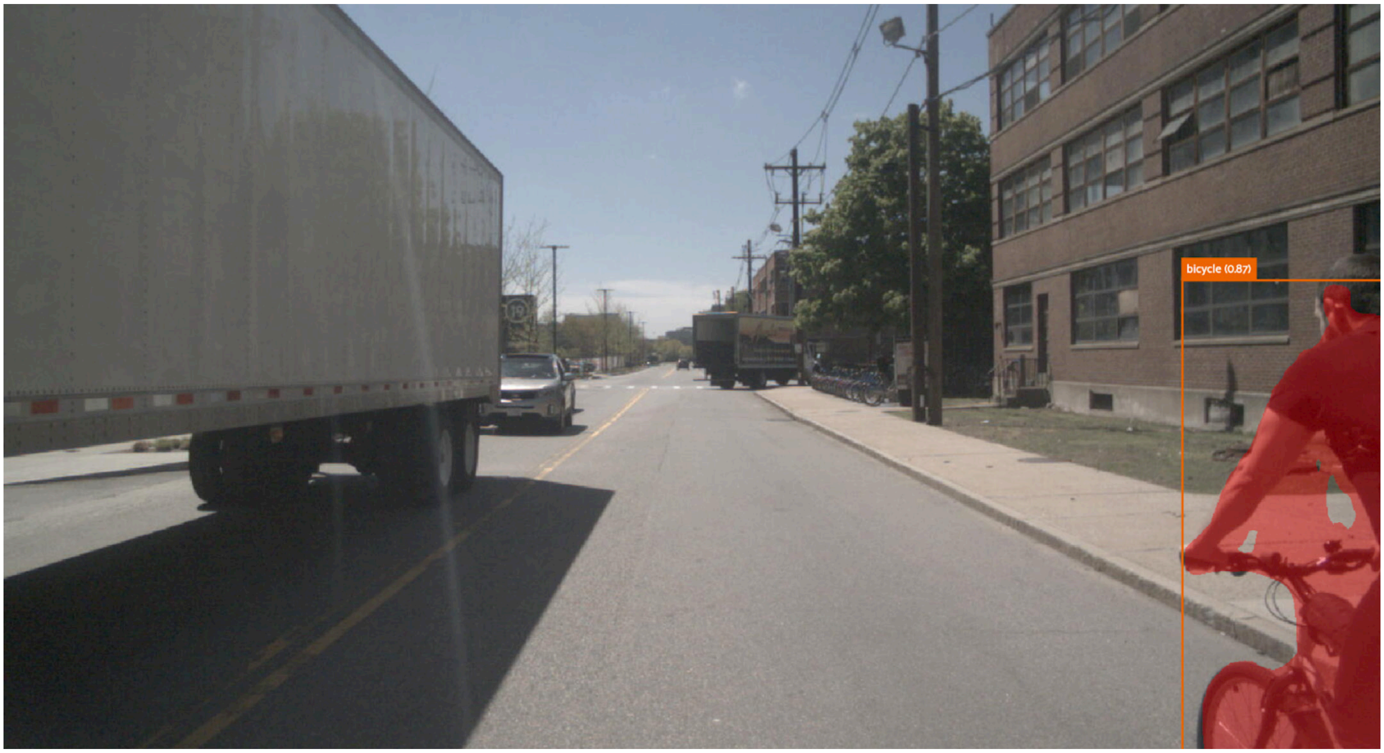


Figure 10 d): The model pre-trained on synthetic data detects a partially occluded cyclist at a close distance from the ego.

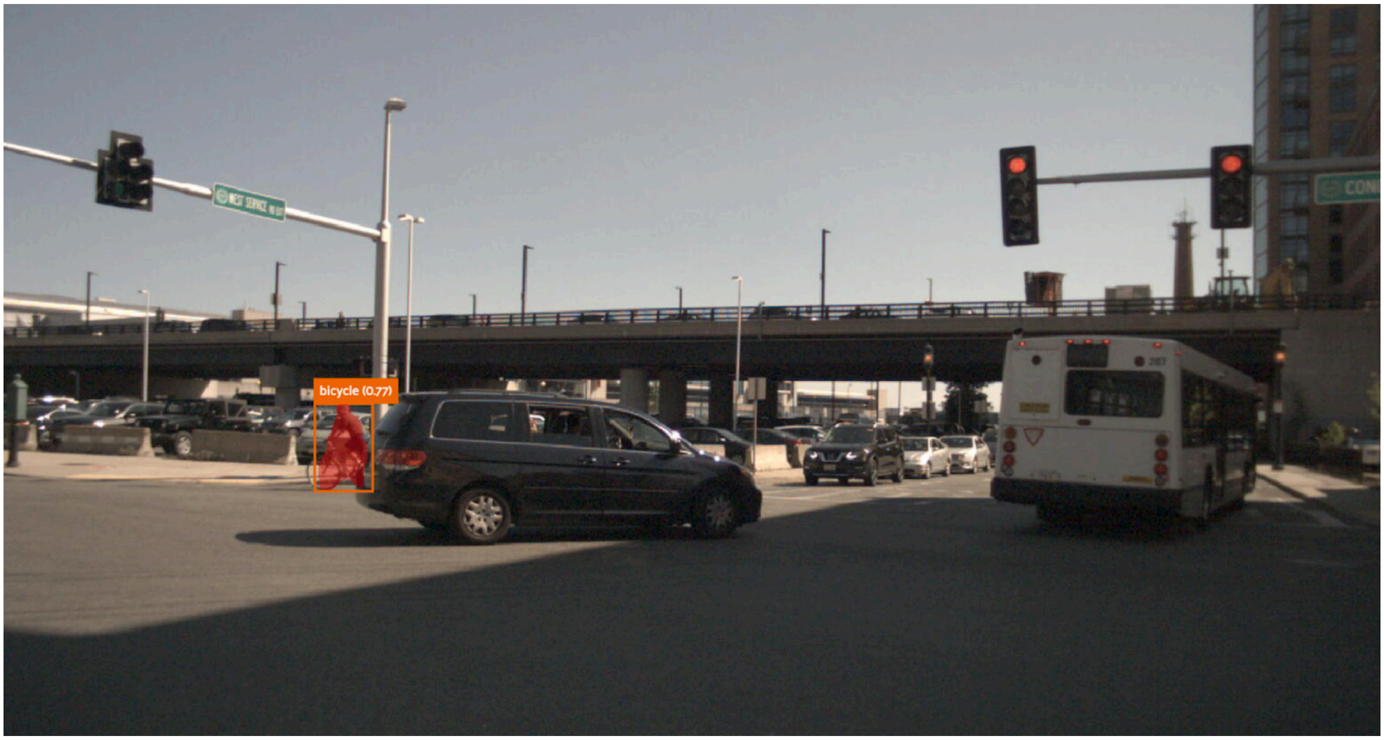


Figure 10 e): The model pre-trained on synthetic data detects a partially occluded cyclist at a distance.

VII. Conclusion

This case study demonstrates that synthetic data is a useful supplementary tool to real-world datasets when training perception algorithms for autonomous vehicles. When used as a complementary resource in training, synthetic data helps address certain class imbalances by improving a perception model's object detection performance on minority classes such as cyclists.

The highest object detection performance is achieved across all classes when pre-training a perception model on a synthetic dataset and consequently training it on a real-world dataset (fine-tuning without data ablation). Furthermore, performance comparable to that of the baseline model is achieved with 50% of the real-world data when pre-training on the synthetic dataset, suggesting that synthetic data can be used to reduce the need for real-world data collection and labeling.

While real-world data has the highest fidelity and accuracy when training perception systems, synthetic data can be used to either improve model performance on rare classes and long-tail events or reduce the cost of collecting additional real-world data.

VIII. Citations

[1] Caesar, Holger, et al. "nuScenes: A multimodal dataset for autonomous driving." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[2] Cai, Zhaowei, and Vasconcelos, Nuno. "Cascade r-cnn: High quality object detection and instance segmentation." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[3] Lin, Tsung-Yi, et al. "Focal loss for dense object detection." *Proceedings of the IEEE international conference on computer vision*, 2017.

[4] Tobin, Josh, et al. "Domain randomization for transferring deep neural networks from simulation to the real world." *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017.

[5] Prakash, Aayush, et al. "Structured domain randomization: Bridging the reality gap by context-aware synthetic data." *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019.

Applied Intuition

applied.co