# Coyote: A Dataset of Challenging Scenarios in Visual Perception for Autonomous Vehicles

**Suruchi Gupta**[1] , **Ihsan Ullah**[2] , **Michael G. Madden**[1]*

[1]School of Computer Science, National University of Ireland Galway, Galway, Ireland
[2]CeADAR Ireland's Center for Applied AI, University College Dublin, Dublin, Ireland
{s.gupta9, michael.madden}@nuigalway.ie, ihsan.ullah@ucd.ie

## Abstract

Recent advances in Artificial Intelligence have immense potential for the realization of self-driving applications. In particular, deep neural networks are being applied to object detection and semantic segmentation, to support the operation of semi-autonomous vehicles. While full Level 5 autonomy is not yet available, elements of these technologies are being brought to market in advanced driver assistance systems that provide partial automation at Level 2 and 3. However, multiple studies have demonstrated that current state-of-the-art deep learning models can make high-confidence but incorrect predictions. In the context of a critical application such as understanding the scene in front of a vehicle, which must be robust, accurate and in real-time, such failures raise concerns; most significantly, they may pose a substantial threat to the safety of the vehicle's occupants and other people with whom the vehicle shares the road.

To examine the challenges of current computer vision approaches in the context of autonomous and semi-autonomous vehicles, we have created a new test dataset, called Coyote[1], with photographs that can be understood correctly by humans but might not be successfully parsed by current state-of-the-art image recognition systems. The dataset has 894 photographs with over 1700 ground-truth labels, grouped into 6 broad categories.

We have tested the dataset against existing state-of-the-art object detection (YOLOv3 & Faster R-CNN) and semantic segmentation (DeepLabv3) models to measure the models' performance and identify situations that might be a source of risk to transportation safety. Our results demonstrate that these models can be confused for various adversarial examples resulting in lower performance than expected: YOLOv3 achieves an accuracy of 49% and precision of 62%, while Faster R-CNN achieves an accuracy of 52% and precision of 60%.

## 1 Introduction

An Autonomous Vehicle (AV) perceives its environment using sensors such as radar, sonar, GPS, and cameras, and uses an advanced control system to identify an appropriate navigation path [Janai *et al.*, 2017]. For this, AV architectures make use of the field of computer vision to interpret and understand their visual inputs. Attempts to provide computers with an understanding of visual components around them dates back to the 1960s [Papert, 1966]. Before the emergence of Convolutional Neural Networks (CNNs) [Krizhevsky *et al.*, 2017], traditional algorithms were used to extract edges and identify shapes. These extracted structural features were then used to identify elements of an image [Szeliski, 2011].

Although researchers have reported that the performance of modern computer vision system approaches human-level performance [Russakovsky *et al.*, 2015], other research studies have conversely demonstrated that images with small perturbations or minor features that should be irrelevant can adversely affect performance [Hendrycks *et al.*, 2019; Nguyen *et al.*, 2015; Szegedy *et al.*, 2014]. Such images, known as adversarial examples, can occur naturally [Hendrycks *et al.*, 2019] or be user-constructed [Nguyen *et al.*, 2015]. In this paper, we explore similar ideas, focusing specifically on the domain of computer vision for (semi-)autonomous vehicles.

**Our Contributions:**

1. We have compiled and annotated a dataset from publicly available images of real-world photographs that are easily understood by humans but might not be parsed successfully by computer vision systems.
2. We have used this dataset to evaluate the performance of current state-of-the-art CNN-based computer vision systems, to identify challenging scenarios that can lead to erroneous performance in self-driving applications.
3. We have analysed the affects of these scenarios on the performance of autonomous vehicles.
4. We have considered the key risks associated with these challenging scenarios, and proposed some mitigations. As we will note, improvements to computer vision models, or using them in combination with other sensor sys-

*Contact Author

[1] https://github.com/Suruchidgupta/UniversalAdversarialChallenges-AutonomousVehicles

tems, can reduce the risk but may not remove the risk entirely.

## 2 Related Work

There are multiple computer vision datasets for autonomous vehicle applications. Cameras from an autonomous driving platform were used to acquire 13k images for the KITTI dataset [Geiger *et al.*, 2013], where scenarios include road, city, residential, campus, etc. KITTI is often used for evaluation only, due to its limited size [Janai *et al.*, 2017]. The Cityscapes dataset [Cordts *et al.*, 2016] contains pixel-level semantic labelling for 25k images related to urban scenarios from 50 cities. It has more detailed annotations than KITTI but does not cover as many scenarios. The ApolloScape dataset [Huang *et al.*, 2020] provides 140k labelled images of street views for lane detection, car detection, semantic segmentation, etc., and is intended to enable performance evaluation across different times of day and weather conditions [Janai *et al.*, 2017]. The WoodScape dataset for autonomous cars [Yogamani *et al.*, 2019] provides 10k images from 4 dedicated fisheye cameras with semantic annotations for 40 classes.

Since images in our dataset include a wide variety of objects that might not be associated with vehicles, we use the Microsoft COCO dataset (Common Objects in Context) [Lin *et al.*, 2014] as the basis for our trained classifiers and for comparative evaluation. COCO contains 328k images with 80 labels of commonly available objects in their surroundings, that could be recognised by a young child.

With the increasing use of deep neural network (DNN) models for image processing, there have been multiple analyses of how these models can be attacked. Experiments have shown that small but carefully chosen perturbations in data can significantly decrease the performance of models. These perturbations, known as adversarial examples, can either be naturally-occurring unseen scenarios [Hendrycks *et al.*, 2019] or user-constructed [Nguyen *et al.*, 2015] [Szegedy *et al.*, 2014] to induce mistakes. Szegedy et al. [2014] used a pre-trained network and derived perturbations specific to an image by making small adjustments to specific pixels that are not noticed by the human eye but result in the images being misclassified by the network. Conversely, Nguyen et al. [2015] generated random images that do not appear recognisable to the human eye but are classified as objects with high confidence by a DNN.

In [Hendrycks *et al.*, 2019], a set of real-world images were collected that contain object classes on which DNNs are trained, but are challenging for DNNs to classify because of features such as texture, shape and background. They also added images with unseen classes and found that the DNN model made high-confidence incorrect classifications, rather than having low confidence in recognising unseen classes, raising further concerns about the reliability of current DNN models for handling unseen examples. Some images from [Hendrycks *et al.*, 2019] are shown in Fig. 1.

Other work has focused on the use of adversarial examples to improve upon existing state-of-the-art models by harnessing these examples to build new DNN models that are resistant to adversarial attacks [Xie *et al.*, 2020; Madry *et al.*, 2018]. Xie et al. [2020] have used adversarial examples as a sample space while training the model to prevent over-fitting and improve the overall performance of the model. Madry et al. [2018] have laid out optimization techniques to handle "the first-order adversary" and building adversary robustness into models for accurate classification results. In our work, we are not concerned with advarsiarial attacks through image modification, but with problems arising from "edge cases" that might not be well covered in training sets but that will occur in the real world.

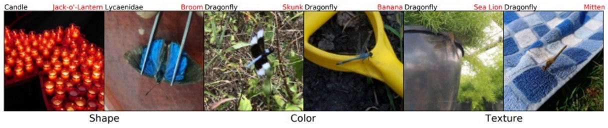


Figure 1: Natural Adversarial Examples from [Hendrycks *et al.*, 2019]

Of more direct relevance to our work, there are a few adversarial datasets for self-driving applications. For instance, WildDash [Zendel *et al.*, 2018] is a test dataset containing 1800 frames addressing the natural risks in images like distortion, overexposure, windscreen, etc. The dataset considers the road conditions from diverse geographic locations, weather and lighting conditions to reduce the bias in training. Similarly, the Adverse Conditions Dataset with Correspondences (ACDC) for semantic driving scene understanding [Sakaridis *et al.*, 2021] studies the effects of four conditions: fog, nighttime, rain, and snow on semantic segmentation using a set of 4006 images. The dataset includes a normal-condition image for each adverse-condition image to identify the challenges with changing weather and lighting conditions and it aims to be used in conjunction with the existing datasets to improve the model performance under the aforementioned conditions. Additionally, the FishyScapes dataset [Blum *et al.*, 2019] tries to place anomalous objects in front of the vehicle and evaluates them on various state-of-the-art semantic segmentation models. It uses the images from the CityScapes [Cordts *et al.*, 2016] dataset, overlays the objects at random distance and sizes to study the model performance in presence of anomalous objects.

While those datasets include high-quality pixel-level semantic segmentations, they do not cover the diverse range of scenarios covered in our Coyote dataset. We hope that the Coyote dataset can form a basis for testing computer vision for autonomous vehicles in edge-case scenarios. Moreover, the image collection can be extended so that a larger version could be used to train computer vision systems that yield better performance and resilience to edge-cases and adversarial attacks, thereby improving automotive safety.

## 3 Overview of the Coyote Dataset

Our Coyote dataset consists of 894 photographs with over 1700 ground-truth labels, grouped into 6 broad categories, as briefly outlined in Fig. 2 and described in the following subsections. We have named this dataset after the cartoon character Wile E. Coyote, who sometimes used realistic murals to mislead the Road Runner. We have chosen photographs that we consider to be easily understood correctly by humans, but not necessarily parsed correctly by current state-of-the-art image recognition systems.

### 3.1 Collection Methodology

Initially, we collected a sample image set that might potentially influence the performance of autonomous vehicles and configured the state-of-the-art object detection models. We evaluated these images on state-of-the-art object detection models and employed an iterative approach and use the outcomes to refine the collection process iteratively. As we collected images, we organised them into categories: Street Signs; Vehicle Art and Textures; Art in Surroundings and Murals; Parking Spaces; On-road Scenarios; and Advanced Scenarios. The images contain either the front or side view of the objects. In almost all cases, the images are un-edited, but we cropped 3 images in the dataset to reduce the background noise in them. The images collected are of different sizes and aspect ratios.

**General Data Protection Regulation Considerations:** All images selected for inclusion in the dataset are publicly available, free for distribution and are labeled for reuse under the Creative Commons license. We avoided many other images because of copyright restrictions.

### 3.2 Art in Surroundings and Murals

Visual art dates back to ancient civilisations and was used as an effective way of communication without using words. Streets and their surroundings, across the world, witness this form of art; some consider it to be a means of communication whereas others consider it vandalism. Either way, an autonomous vehicle must distinguish works of art from reality. Hence, this category aims to identify art that exists near roads and that might deteriorate self-driving applications' performance.

**Re-creation of a road scenario:** Some art painted on walls depicts streets with components like cars, traffic lights, cycles, pedestrians, etc. For example, Fig. 3(a) is a mural based on the Beatles' Abbey Road album, which might be interpreted as a roadway rather than a wall.

**False identification of risks in surroundings:** Some murals contain elements that may be misidentified as a source of threat for the occupants, e.g. pictures of accidents, wild animals, natural calamities, etc. The mural in Fig. 3(b)) might be misinterpreted as a crashed car.

**Art representing road objects:** Sculptures and other artworks may depict objects typically found on the road, such as cars, trucks, motorbike, etc. There is a risk that artworks such as the one in Fig. 3(c)) can be confused with an actual car.

### 3.3 Vehicle Art and Textures

Most vehicles conform to standard makes and colours. However, some have unusual artwork or textures, either for artistic reasons or for commercial branding. This category contains images of vehicles that are unusual and as such may be challenging or autonomous vehicle object recognition.

**Vehicles disguised as other objects:** This category includes images of vehicles camouflaged as different objects such as a shoe, telephone, animal (cat, peacock, dragon), or adorned with flowers, skulls, and other designs. Fig. 3(d) shows a car disguised as a cat. Interestingly, Houston hosts an Art Car Parade every year to showcase unique car designs; more examples can be found on their website.

**Vehicles with Textures:** Some companies and individuals use texture as a medium to advertise their brand or decorate their vehicles. Vehicles are either covered with a specific pattern such as grass, cow patches, tiger prints, etc. or by small assorted patterns to create a unique effect on the automobile body (Fig. 3(e)). Alternatively, some vehicles can also have a scene painted on their body, such as an brand image, a graphic art book image, or a movie scene.

**Custom Built Vehicles:** Some vehicles are uniquely designed as 'positional goods' to have distinguishing features, such as custom prints, solar panels, dual engines, etc. As shown in the Fig. 3(f), some of these vehicles are hybrids of different automobiles; for instance, the car that looks similar to a helicopter might hinder object recognition.

### 3.4 On-Road Scenarios

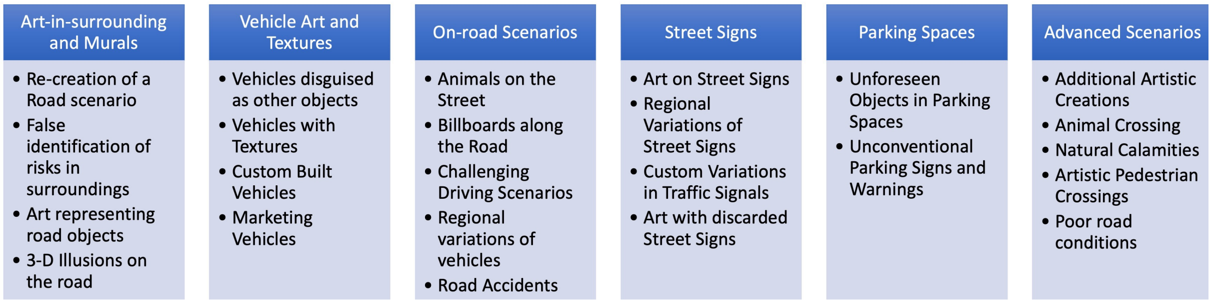
Computer vision systems for autonomous vehicles are trained on datasets relevant to the road, which contain road objects and scenarios to help the model identify on-road components. However, scenarios across the world are so diverse that it is challenging to ensure that all possible scenarios are included in training datasets. The images in this category are unusual but realistic on-road scenarios that might be challenging for autonomous vehicle object recognition.

**Animals on the Street:** As humans expand our habitable land, there are multiple places where encounters with animals on roads is not unusual. Hence, images in this category show different species of animal, wild and domestic, wandering across the streets in rural and urban scenarios (e.g. Fig. 3(g)). Unlike other categories such as murals and those involving other vehicles, the behaviour of animals is difficult to predict and hence can be difficult for the autonomous vehicle to handle.

**Billboards along the Road:** Billboards are commonly seen along roads. Although they should not interfere with an autonomous vehicle's operation, their presence can potentially confuse it. The image in Fig. 3(h) shows an election poster. If an autonomous vehicle mistakes a poster for a real human, it might apply emergency brakes, leading to erratic driving behaviour.

**Challenging Driving Scenarios:** When employed in the physical world, autonomous vehicles are subjected to different lighting conditions and varying weather conditions (fog, rain, snow, etc.) throughout the year. Hence, they must be aware of different weather conditions and their resulting impact on the surroundings. The image in Fig. 3(i) is an example of the change in the weather condition. For instance, if the autonomous vehicle does not correctly identify the objects in low visibility or in varying weather conditions, it might not modify its behaviour accordingly. This category also includes images of regional variations of vehicles (tricycles for public transport, cargo bicycles for delivery, etc.), challenging roads scenarios (such as mountains, valleys, etc.), and images of extreme situations such as accidents, to evaluate how au-

https://www.thehoustonartcarparade.com/



Art-in-surrounding and Murals
- Re-creation of a Road scenario
- False identification of risks in surroundings
- Art representing road objects
- 3-D Illusions on the road

Vehicle Art and Textures
- Vehicles disguised as other objects
- Vehicles with Textures
- Custom Built Vehicles
- Marketing Vehicles

On-road Scenarios
- Animals on the Street
- Billboards along the Road
- Challenging Driving Scenarios
- Regional variations of vehicles
- Road Accidents

Street Signs
- Art on Street Signs
- Regional Variations of Street Signs
- Custom Variations in Traffic Signals
- Art with discarded Street Signs

Parking Spaces
- Unforeseen Objects in Parking Spaces
- Unconventional Parking Signs and Warnings

Advanced Scenarios
- Additional Artistic Creations
- Animal Crossing
- Natural Calamities
- Artistic Pedestrian Crossings
- Poor road conditions

Figure 2: Broad categories of images in Coyote dataset

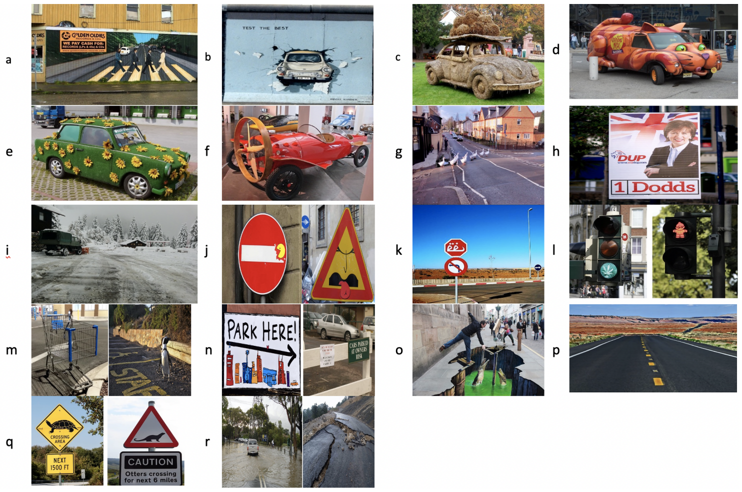


Figure 3: (a) Sample murals re-creating road scenarios; (b) Paintings on road depicting safety threat; (c) Sculptures that resemble road objects such as cars and trucks; (d) Motor vehicles camouflaged as other objects; (e) Textures to decorate vehicles; (f) Custom built motor vehicle; (g) Photos of animals on roads; (h) Road-side signs with pictures of humans; (i) Challenging road scenarios for AVs; (j) Street signs modified by artists; (k) Street signs in Arabic; (l) Custom traffic signals on the road; (m) Images showing presence of unseen objects in the parking space; (n) Non-standard signs and warnings; (o) 3-dimensional illusions that may challenge semantic segmentation; (p) Additional art that might act as adversarial examples; (q) Examples of animal crossing signs; (r) Natural events

tonomous vehicles handle such scenarios. Other datasets such as the Yamaha-CMU Off-Road dataset (YCOR) [Maturana *et al.*, 2017] and the PASCAL-VOC dataset also includes some extreme weather scenarios but does not include other scenarios presented in the Coyote dataset.

### 3.5 Street Signs

Street signs have an important function in guiding and providing instructions to road users, but street signs that are misunderstood by autonomous vehicle would have potentially an adverse effect. This category includes regional variations of street signs across the world and modifications made by artists to street signs.

**Art on Street Signs:** Some artists have modified the existing road sign elements to create interesting variations. These variations generally do not affect humans' ability to recognise them but may be more challenging for autonomous vehicles. For example, if the modified speed bump sign or stop sign in Fig. 3(j) is ignored, the autonomous vehicle may fail to reduce its speed.

**Regional Variations of Street Signs:** While street signs are generally standardised within a region, there are many variations across regions. Most regions have street signs in the regional language (e.g. Fig. 3(k)). While autonomous vehicles sold in a region would be configured to handle these regional variations, they could cause problems for vehicles that travel between regions or that are imported by the owner into an unfamiliar region.

**Custom Variations in Traffic Signals:** While curating images relating to street signs, we encountered some custom traffic signals. Instead of standard circular lights, they may have custom figures in red and green colours. Signals such as Fig. 3(l) are easily understood by humans but deviate from what autonomous vehicles may have been trained on.

This category also contains images of a large mosaic artwork created from discarded street signs. Identifying any of the street signs in the art might lead to an unexpected outcome by the vehicle. The COCO dataset only identifies the stop sign; a more comprehensive study with a domain-centred dataset can provide insights into the effects of these street signs on autonomous vehicles.

### 3.6 Parking Spaces

This category covers parking spaces and their environments. It includes examples of animals or objects in the parking space and non-standard environments such as rural settings without the standard parking boxes.

**Unforeseen Objects in Parking Spaces:** The images in this category include miscellaneous objects in the parking spaces, such as animals, shopping carts, etc. (e.g. Fig. 3(m)). The presence of unidentified objects like shopping carts (left) and animals (right) in the parking spaces might lead to misjudgment by the autonomous vehicle.

**Unconventional Parking Signs and Warnings:** There are cases where authorities display warnings/notices or customised parking/no-parking signs that are not easily interpreted. The sample image on the left in Fig. 3(n) shows one of such unconventional parking notice, while the image on the right shows a warning sign regarding the icy car paths ahead, that would require significant natural language processing to interpret.

### 3.7 Semantic Segmentation

While collecting photos in the category of *Art in surrounding and Murals*, we found images that create an illusion of 3D space. To examine how an autonomous vehicle might parse these scenarios, we have evaluated these images on a semantic segmentation model. The images depict multiple scenarios, for instance, Fig. 3(o) shows a painting of a hole with water and the presence of wild animals, etc. Such images could result in the vehicle failing to proceed and blocking the road if access to the road is damaged and there is no way to go forward.

### 3.8 Advanced Scenarios

The COCO dataset covers only 80 commonly available object classes, and not all relevant scenarios can be covered by these 80 classes. Hence, the Advanced Scenarios contain images of objects or scenarios that may be unrecognisable for a model trained on COCO or a similar dataset.

**Additional Artistic Creations:** Section 3.2 described scenarios where art can potentially mislead autonomous vehicles. This category contains art images with objects that are not in the COCO dataset classes, which may be even harder to handle. The image in Fig. 3(p) shows a painting of a road; this might lead a vehicle to incorrectly drive ahead, compromising occupants' safety.

**Animals Crossing:** Different forms of animal crossing signs are used to ensure all intended users' safety. It may be challenging for autonomous vehicles to identify all such signs and also understand details like the distance over which the sign applies. This category includes a variety of animal crossing signs from across the world, e.g. Fig. 3(q).

**Natural Calamities:** With the ever-changing weather and environmental conditions, natural calamities are a risk that cannot be eliminated. These incidents often severely damage the transportation infrastructure around us. This category includes examples of natural calamities. Fig. 3(r) shows road damage that occurred as a result of a flood (left) and an earthquake (right).

### 3.9 Summary of Dataset

The curated dataset contains a total of 894 images across six categories. The number of images in each category is given in Table 1. The highest number of objects in the Coyote dataset are (in descending order) person, car, truck, bicycle, motorbike, traffic light, bus, stop sign, train, bird, cow, and umbrella, followed by others with fewer than ten occurrences. We have released the images and our ground truth labels for research purposes. Images have unique file names. An accompanying spreadsheet provides manually-annotated ground truth, comprising the file name and a count of objects of each class in the image, using the same set of classes as used in the COCO dataset. The database also contains an appendix with a list of links to the sources of all images.

## 4 Experiments

### 4.1 Experimental Methodology

As discussed in Section 1, this project employs state-of-the-art object detection and semantic segmentation models to test the collected road scenarios. The models used are pre-trained on the COCO dataset to identify 80 common object classes in the surroundings and are not altered during this experiment. The experiments are conducted on a MacBook Pro running macOS Catalina version 10.15.6.

After collecting the initial sample set, we configured the state-of-the-art models to run inference. The threshold for the Object Detection models is set as 70%. Subsequently, we labeled all the object classes in the images manually using the output classes in the MS-COCO dataset to generate the ground truth for the data. Finally, we used the generated ground truth data for the implementation of the evaluation metrics and summarised the results to infer the overall outcome of the project.

To compare the results of the Coyote dataset with benchmark datasets, we have used the MS-COCO 2017 Validation set and a random subset of 1715 images from the KITTI dataset testing set. The images from these datasets are tested in the same setup as the Coyote dataset. The KITTI dataset has different class labels from those of MS-COCO, so we mapped them to the closest matching categories in MS-COCO (e.g. Pedestrian maps to Person) and evaluated them with the same set of metrics, to enable valid comparisons.

**Metrics:** To evaluate the performance of the models in line with what is done in other work such as [Hung *et al.*, 2020; Benjdira *et al.*, 2019], we used the following metrics: True Positives (TP), False Positives (FP), False Negatives (FN), Accuracy (Acc), precision (Prec), recall (Rec), and F1-score (F1). As usual, True Negatives are not counted since it is not useful to note that an object class does not exist in an image and that object class was not detected; since MS-COCO has 80 classes, we would have a huge number of TNs for every image. For simplicity, the ground truth labels used in the Coyote dataset identify the number of occurrence for each of the output classes and the present version does not include bounding boxes. Hence, we cannot compute mAP or IoU for the Coyote dataset.

### 4.2 Results with YOLOv3 & Faster R-CNN

**YOLOv3:** YOLOv3 internally uses a darknet architecture trained on MS-COCO. It provides a robust single-shot approach to object detection and is known to be at par with other

https://github.com/Suruchidgupta/UniversalAdversarialChallenges-AutonomousVehicles

https://pjreddie.com/darknet/yolo/

Table 1: Summary of type and number of images in the dataset.

| Category Name | Advanced Scenarios | Art-in-surrounding and Murals | On-road Scenario | Parking spaces | Street Signs | Vehicle Art and Textures | **Total** |
|---|---|---|---|---|---|---|---|
| Number of Images | 99 | 226 | 210 | 41 | 99 | 219 | **894** |

Table 2: Cumulative results for Faster R-CNN (FRCNN) model vs. YOLOV3 on Coyote, KITTI, and MS-COCO datasets. The overall result show a significant drop in precision for the Coyote dataset. Red color shows lowest performance among the three datasets in the column

| **Category Name** | **Accuracy** | | **Precision** | | **Recall** | | **F1-score** | | **True Positive** | | **False Positive** | | **False Negative** | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | FRCNN | YOLOv3 | FRCNN | YOLOv3 | FRCNN | YOLOv3 | FRCNN | YOLOv3 | FRCNN | YOLOv3 | FRCNN | YOLOv3 | FRCNN | YOLOv3 |
| **On-road Scenario** | 0.64 | 0.61 | 0.77 | 0.88 | 0.79 | 0.67 | 0.78 | 0.76 | 550 | 467 | 161 | 61 | 149 | 232 |
| **Art in surroundings & Murals** | 0.15 | 0.24 | 0.15 | 0.25 | 0.86 | 0.78 | 0.26 | 0.38 | 84 | 76 | 460 | 223 | 14 | 22 |
| **Street Signs** | 0.56 | 0.5 | 0.63 | 0.8 | 0.84 | 0.58 | 0.72 | 0.67 | 74 | 51 | 43 | 13 | 14 | 37 |
| **Parking spaces** | 0.78 | 0.7 | 0.82 | 0.91 | 0.94 | 0.75 | 0.88 | 0.82 | 108 | 86 | 24 | 8 | 7 | 29 |
| **Vehicle Art and Textures** | 0.59 | 0.61 | 0.72 | 0.9 | 0.76 | 0.65 | 0.74 | 0.75 | 542 | 460 | 212 | 49 | 167 | 249 |
| **Coyote Total** | 0.52 | 0.49 | 0.60 | 0.62 | 0.79 | 0.71 | 0.68 | 0.66 | 1358 | 1211 | 900 | 739 | 351 | 498 |
| — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| **MS-COCO 2017 Val Set** | 0.53 | 0.46 | **0.91** | 0.79 | 0.56 | 0.53 | 0.69 | 0.63 | 11211 | 10668 | 1069 | 2915 | 8977 | 9520 |
| **KITTI Testing Subset** | **0.76** | 0.63 | 0.86 | 0.81 | **0.87** | 0.74 | **0.86** | 0.77 | 3778 | 3213 | 604 | 735 | 564 | 129 |

models such as Faster R-CNN. Fig. 4 shows some cases from the Coyote dataset where YOLOv3 performs correctly. In the top-left and bottom-right images, the model distinguishes between the billboard/mural and the person with confidences of 99%. It identifies the car decorated with fruit (top-right) and the altered stop sign (bottom-left) with confidence values of 87% and 95%.

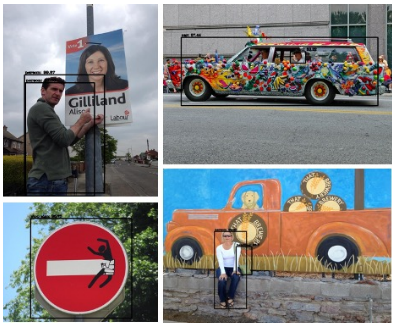


Figure 4: YOLOv3: some successful examples. (Top-left) Man standing beside a billboard; (top-right) Car decorated with fruit; (bottom-left) Art on a stop sign; (bottom-right) Woman sitting in front of a truck mural.

However, Fig. 5 shows other examples where YOLOv3 fails. In the top-left image, a person-like sculpture is mounted on an old bicycle, and the model identifies it as a person with 95% confidence. In the top-right image, parking spaces for bikes are bike-shaped, causing false detection of a bicycle with 93% confidence. The stop sign in the centre-left and the car in the bottom-left are not detected because of the art on the sign and the grass texture on the car. Additionally, the murals in the centre-right and bottom-right images are incorrectly identified as real objects.

The overall performance for YOLOv3 across all categories is provided below in Table 2. The high FP value in the *Art in Surroundings and Murals* category indicates that multiple objects in the art are identified as real objects. Conversely, in other categories, the high number of FNs indicates that there are objects present that the model cannot find. YOLOv3's overall accuracy on the Coyote dataset is low at 49%, with precision=62%, recall=71%, and F1-score=66%. The statistics in the table indicate that the images are challenging for YOLOv3. **Faster R-CNN:** The Faster R-CNN model trained on MS-COCO is built on the ResNet101 architecture with 1024x1024 resolution. Faster R-CNN identifies more details in the images than the YOLOv3 and hence, provides good results for many images in the dataset.

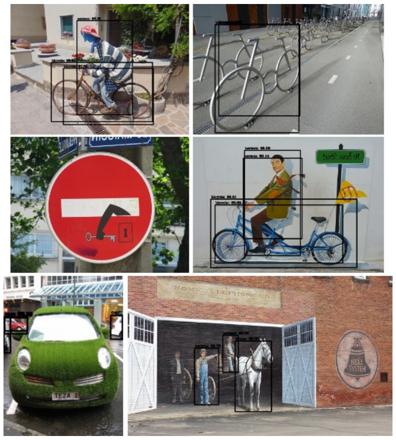


Figure 5: YOLOv3: some unsuccessful cases. (Top-left) Sculpture mistaken for a person on bicycle; (Top-right) bike parking confused for bicycle; (centre-left) stop sign not detected; (centre-right) mural with a person on a bicycle classified as real; (bottom-left) undetected car with grass texture; (bottom-right) mural containing cart and people identified as real.

Fig. 6 top-left shows the successful identification of a bicycle mounted on the front of a bus. The top-right and bottom-right images show the instances where YOLOv3 fails but Faster R-CNN successfully identifies the stop sign and the grass-textured car. The bottom-left image shows that a mural with a cyclist does not confuse the model. Some unsuccessful results for the Faster R-CNN model are shown in Fig. 7. The top-left and centre-left images show that Faster R-CNN identified the objects from billboards and murals as real with high confidence (98% and 99%, respectively). The top-right image shows that the decorated car is not missed while others are successfully classified. The model misclassifies the car on the centre-right as a cake with 97% confidence and the shopping cart as a bicycle with 91% confidence. The bottom-right image shows a sculpture made from discarded street signs, but the model identifies some as actual stop signs and misclassifies the arrows as a parking meter with 90% confidence.

As shown in Table 2, the overall performance of Faster R-CNN is not strong. Like YOLOv3, in the *Art in Surroundings and Murals* category has a high FP value, implying that the

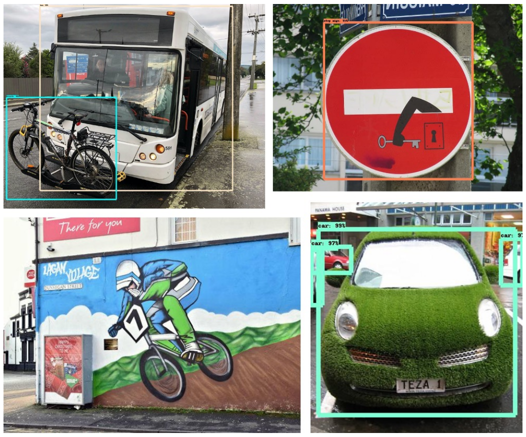


Figure 6: Faster R-CNN: sample successful results. (Top-left) Bus with bicycle mounted in front; (top-right) modified stop sign; (bottom-left) mural with a cyclist; (bottom-right) car with grass texture.

model is confused by the painting and murals. Faster R-CNN has a high FP level for the *Vehicle Art and Texture* category also. In the other categories, there are also significant FN levels, though they are not quite as high as YOLOv3.

The number of TPs for Faster R-CNN is significantly higher than that of YOLOv3, leading to fewer FNs. However, Faster R-CNN has a very high number of FPs, which reduces the model's overall performance. Hence, its accuracy is very low at 52%, precision=60%, recall=79%, and F1-score=68%. On the whole, the Coyote dataset affects the performance of both Faster R-CNN and YOLOv3 models.

**Comparison with KITTI and MS-COCO:** We performed further experiments to evaluate Faster R-CNN and YOLOv3 on the MS-COCO 2017 validation set and a subset of the KITTI dataset. The objective was to determine whether the Coyote dataset is indeed more challenging than existing benchmark datasets. The results in Table 2 show that the precision of the DNN models on the Coyote dataset drops by 31% and 26% with Faster-RCNN and 17% and 8% with YOLOv3, relative to MS-COCO and KITTI, respectively. In addition, the F1-scores are lower on the Coyote dataset. Hence, a vehicle that uses such models would be at risk of accidents when it faces real-world edge-case examples as exemplified by the Coyote dataset.

### 4.3 Semantic Segmentation with DeepLabv3

Semantic segmentation is an extension of the proposed project. DeepLabv3 [Chen *et al.*, 2017] is pre-trained on the MS-COCO dataset and uses the colour map from the PASCAL VOC dataset with 21 output labels (background + 20 PASCAL VOC labels). The DeepLabv3 model used for this project operates on the MobileNetv2 architecture with a depth multiplier of 0.5 [Sandler *et al.*, 2018]. We apply it to a small collection of 19 images and evaluate the results manually.

The output of semantic segmentation for some images show correct results, where the model is not affected by the presence of confusing art around them. For example, Fig. 8(top) shows art creating an illusion of ice blocks and a person fishing. DeepLabV3 correctly segments the people in the image is not affected by the art. In Fig. 8(bottom), people are standing beside a painting of a bicycle and road with a tearing illusion. The model classifies the cyclist and the bicycle as real, along with the real people standing beside them.

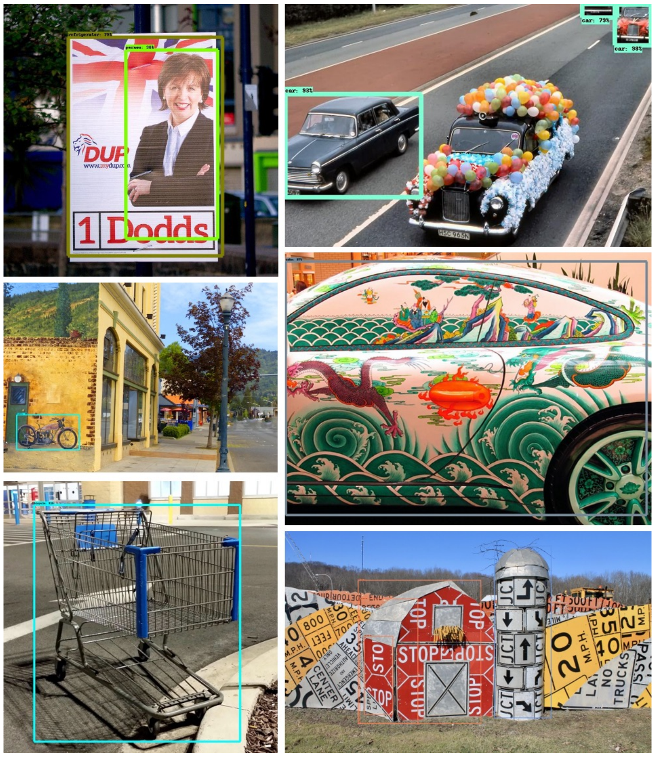


Figure 7: Faster R-CNN: sample unsuccessful results. (Top-left) Photo in poster identified as real person; (top-right) car not detected; (centre-left) painting of motorcycle identified as real; (centre-right) decorated car identified as cake; (bottom-left) Shopping cart identified as bic [illegible] eet signs.

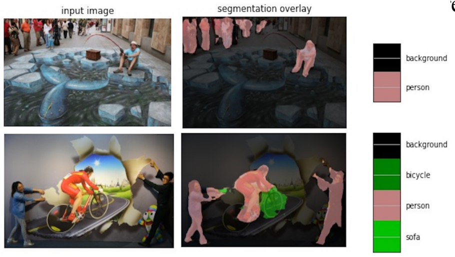


Figure 8: DeepLabv3 semantic segmentation samples. (Top) Illusion of ice blocks on the road and a person fishing; (bottom) people beside an illusion of a road and cyclist tearing through a wall.

On analysing the results for all 19 images, we observe that the reason why the images in Fig. 8(top) and are correctly segmented can be explained by the limited number of output classes in the PASCAL VOC dataset, which do not include ice, for example. The most common failures in semantic segmentation are in classifying art as real objects; this can make the autonomous vehicles susceptible to failure in presence of 3-D illusions.

## 5 Conclusion

### 5.1 Analysis of Results

In this paper, we have presented a new publicly available test dataset, called Coyote, of real-world photographs that are easily understood by humans but might not be successfully parsed by computer vision systems, along with manual annotations of objects in all of the photographs. The photographs are grouped into six broad categories: (1) art in surrounding and murals; (2) vehicle art and textures; (3) On-road scenarios; (4) street sign; (5) parking spaces; and (6) advanced scenarios.

We have used the Coyote dataset to evaluate the performance of current state-of-the-art CNN-based computer vision systems, to identify challenging scenarios that can lead to erroneous performance in self-driving applications. We have found that the paintings in *Art in surrounding and Murals* category confuse the models the most. Both YOLOv3 and Faster R-CNN models perform worse on this category than any other, with a high number of FPs, showing that the models identify the paintings' components as real objects. In the case of *Street Signs* category, embellishments to street signs and regional variations of road signs degrade the performance of both of the models. The decorated vehicles in the *Vehicle Art and Textures* category show contrasting failure modes for Faster R-CNN and YOLOv3 models. While the YOLOv3 model cannot identify many of the objects in the images (high FNs), Faster R-CNN incorrectly identifies a large number of objects that are not present (high FPs).

The unseen road scenarios presented in the *On-Road scenarios* category yields somewhat better performance for both of the models. However, both models have a high number of FNs, indicating that the models are often unable to identify the objects in these new scenarios. The *Parking Spaces* category, which contains the smallest number of examples, shows the best performance for the models among all of the categories. It is possible that the models are not misled due to the COCO dataset having a limited set of class labels. Nonetheless, this category highlights some interesting adversarial scenarios for self-driving applications. In *Advanced Scenarios* category with 3-D art on roads, DeepLab can correctly segment humans and the background but it fails in some cases when 3-D representations of objects are painted on walls. However, these scenarios could be assessed further using a more comprehensive model trained specifically for road scenarios.

Analysing the aggregate results for both models shows that the Faster R-CNN model captures more details than the YOLOv3 model. This helps the Faster R-CNN model make better inferences; however, it also makes its model more vulnerable to failure from edge-cases.

### 5.2 Risks and Mitigation

The key risk associated with the kinds of errors we identify in this work is that a vehicle may react inappropriately, for example braking sharply to avoid hitting a "person" that is actually an image on the back of a van or preparing to drive ahead when the street ahead is actually a mural. Some strategies for risk mitigation are summarised below.

**Sensor Fusion:** For example, data fused from camera and lidar systems might indicate that an image of a person on the back of a van is flat and therefore cannot be real, thereby reducing false positives. However, if the vision system detects a person ahead with high confidence and lidar does not, must the autonomous vehicle act conservatively to preserve life?

**Common Sense Reasoning:** Work is being done on simulation-based reasoning systems that aim to synthesize understanding of a domain. It is conceivable that such systems could be extended to recognise, for example, that a photograph of the head of a person is not actually a person.

**Better Treatment of Scale:** Depth and scale are important factors for humans in distinguishing real items from artificial ones; e.g. a car decorated to look like a cat. Current CNNs rely to a large extent on textures and patterns and are designed to be scale-invariant.

**Spatio-Temporal Reasoning:** While a single image of a mural may look realistic to a human, when viewed over time from slightly different viewpoints, it rapidly becomes clear that it is 2D. This requires Spatio-temporal reasoning that is beyond the capacity of current systems.

### 5.3 Future Work

There is substantial scope for building on the work presented here. Most obviously, the Coyote dataset can be extended by adding more images. For example, [Mufson, 2017] presents art created by Filip Piekniewski containing scenarios that can confuse autonomous vehicles. In addition, the images could be annotated with bounding boxes and pixel-level semantic annotations. Such fine-grained ground truth will be used in localization of objects detection which is critical information for autonomous vehicles.

Further work will be done on increasing the number of images by increasing the challenging scenarios; e.g. occluded objects such as a person occluded under an umbrella, the field of views from each object, and by capturing it from a camera within a vehicle itself. Additionally, other forms of data(such as LiDAR, multiple Cameras, FishEye, GPS, temporal consistency between frames, etc.) can be brought in context to analyse how much the additional information can support the autonomous vehicles in such challenging scenarios as being done in e.g. nuScenes dataset [Caesar *et al.*, 2020].

## References

[Benjdira *et al.*, 2019] B. Benjdira, T. Khursheed, A. Koubaa, A. Ammar, and K. Ouni. Car detection using unmanned aerial vehicles: Comparison between faster r-cnn and yolov3. In *2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS)*, pages 1–6, 2019.

[Blum *et al.*, 2019] Hermann Blum, Paul-Edouard Sarlin, Juan Nieto, Roland Siegwart, and Cesar Cadena. Fishyscapes: A benchmark for safe semantic segmentation in autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[Caesar *et al.*, 2020] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[Chen *et al.*, 2017] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR*, abs/1706.05587, 2017.

[Cordts *et al.*, 2016] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. pages 3213–3223, 2016.

[Geiger *et al.*, 2013] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *Int. J. Robotics Res.*, 32(11):1231–1237, 2013.

[Hendrycks *et al.*, 2019] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CoRR*, abs/1907.07174, 2019.

[Huang *et al.*, 2020] Xinyu Huang, Peng Wang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The apolloscape open dataset for autonomous driving and its application. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(10):2702–2719, 2020.

[Hung *et al.*, 2020] Goon Li Hung, Mohamad Safwan Bin Sahimi, Hussein Samma, Tarik Adnan Almohamad, and Badr Lahasan. Faster R-CNN Deep Learning Model for Pedestrian Detection from Drone Images. *SN Computer Science*, 1(2):1–9, 2020.

[Janai *et al.*, 2017] Joel Janai, Fatma Güney, Aseem Behl, and Andreas Geiger. Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art. *arXiv e-prints*, page arXiv:1704.05519, April 2017.

[Krizhevsky *et al.*, 2017] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[Madry *et al.*, 2018] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR, Canada, April 30 - May 3*, 2018.

[Maturana *et al.*, 2017] Daniel Maturana, Po-Wei Chou, Masashi Uenoyama, and Sebastian A. Scherer. Real-time semantic mapping for autonomous off-road navigation. In *Field and Service Robotics, 11th International Conference, FSR, Zurich, Switzerland, 12-15 September*, volume 5, pages 335–350. Springer, 2017.

[Mufson, 2017] Beckett Mufson. An engineer is painting surreal scenarios that confuse self-driving cars, 2017.

[Nguyen *et al.*, 2015] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *IEEE conference on computer vision and pattern recognition*, pages 427–436, 2015.

[Papert, 1966] Seymour A Papert. The summer vision project. 1966.

[Russakovsky *et al.*, 2015] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[Sakaridis *et al.*, 2021] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. *arXiv preprint arXiv:2104.13395*, 2021.

[Sandler *et al.*, 2018] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[Szegedy *et al.*, 2014] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR, Banff, AB, Canada, April 14-16,*, 2014.

[Szeliski, 2011] Richard Szeliski. *Computer Vision - Algorithms and Applications*. Texts in Computer Science. Springer, 2011.

[Xie *et al.*, 2020] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L Yuille, and Quoc V Le. Adversarial examples improve image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 819–828, 2020.

[Yogamani *et al.*, 2019] Senthil Yogamani, Ciarán Hughes, Jonathan Horgan, Ganesh Sistu, Padraig Varley, Derek O'Dea, Michal Uricár, Stefan Milz, Martin Simon, Karl Amende, et al. Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In *IEEE International Conference on Computer Vision*, pages 9308–9318, 2019.

[Zendel *et al.*, 2018] Oliver Zendel, Katrin Honauer, Markus Murschitz, Daniel Steininger, and Gustavo Fernandez Dominguez. Wilddash-creating hazard-aware benchmarks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 402–416, 2018.