BDD100K: A Diverse Driving Video Database with Scalable Annotation Tooling

Fisher Yu[1] Wenqi Xian[2] Yingying Chen[1] Fangchen Liu[3] Mike Liao[1]
Vashisht Madhavan[4] Trevor Darrell[1]

[1]UC Berkeley [2]Georgia Institute of Technology [3]Peking University [4]Uber AI Labs

Abstract. Datasets drive vision progress and autonomous driving is a critical vision application, yet existing driving datasets are impoverished in terms of visual content. Driving imagery is becoming plentiful, but annotation is slow and expensive, as annotation tools have not kept pace with the flood of data. Our first contribution is the design and implementation of a scalable annotation system that can provide a comprehensive set of image labels for large-scale driving datasets. Our second contribution is a new driving dataset, facilitated by our tooling, which is an order of magnitude larger than previous efforts, and is comprised of over 100K videos with diverse kinds of annotations including image level tagging, object bounding boxes, drivable areas, lane markings, and full-frame instance segmentation. The dataset possesses geographic, environmental, and weather diversity, which is useful for training models so that they are less likely to be surprised by new conditions. The dataset can be requested at `http://bdd-data.berkeley.edu`.

1 Introduction

Diverse, large-scale annotated visual datasets (ImageNet [8], COCO [14], etc.) have been the driving force behind recent advances in supervised learning tasks in computer vision. Typical deep learning models can require millions of training images to achieve state-of-the-art performance.

For autonomous driving applications, however, leveraging the power of deep learning is not as simple. Existing datasets for autonomous driving are limited in one or more significant aspects, including scene variation, richness of annotation, and geographic distribution. Additionally, models trained on existing datasets tend to overfit specific domain characteristics. To overcome such limitations, we propose, collect, and annotate a new, diverse, and large-scale dataset of visual driving scenes.

Camera-instrumented vehicles are becoming commonplace. As described below, existing platforms allow for large-scale crowdsourcing of dashcam videos, and future vehicles will likely come equipped with streaming-capable cameras. Annotating such massive amounts of data becomes itself a technical challenge. Yet relatively less attention in the literature has been given to the annotation tools that label such data. To achieve rich annotation at scale, we found that existing tooling was insufficient, and therefore develop novel schemes to annotate driving data more efficiently and flexibly than previous methods. Current tools are difficult to deploy at scale and are rarely extensible to new tasks or data-structures.

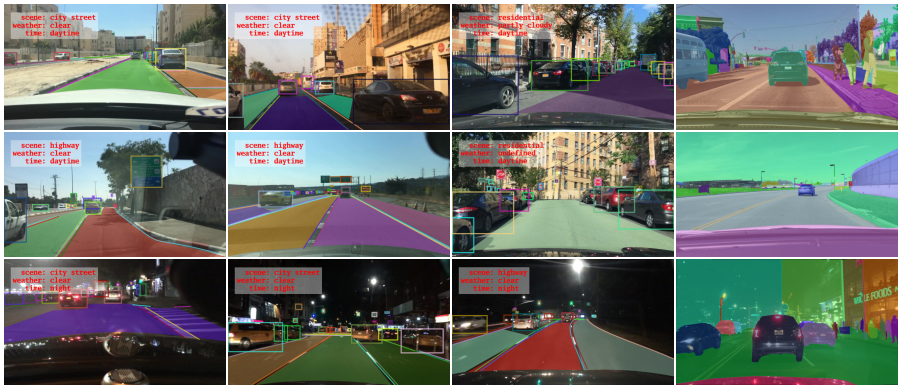


Fig. 1: Overview of our database. Our labeling system can be easily extended to multiple kinds of annotations. With our system, we can label a diverse driving video dataset with several types of annotations: scene tagging, object bounding box, lane, drivable area, and full-frame instance segmentation.

We study how to improve labeling efficiency as well as extensibility, so that an annotation system can be easily adapted to new tasks. In most use cases, there are only three basic types of labels on an image: image-level tagging, bounding box, and polygon annotation. However, the semantic meanings of the annotations can differ dramatically between use cases. Therefore, we build a configurable annotation system that supports these three types of annotations. Moreover, we also introduce improvements to boost labeling efficiency, as described below. We tested our system in annotating real-world videos while extending it to additional types of labels, including drivable area and lane marking annotations. At the same time, we also collected efficient full-frame semantic instance segmentations. Our study shows that we can extend our labeling system to new types of annotation with minimal effort while still being able to label large-scale datasets.

Employing our scalable tooling framework, we have been able to collect and annotate the largest available dataset of annotated driving scenes, comprised of over 100K diverse video clips. Not surprisingly, when evaluating existing algorithms on our newly proposed dataset, we discovered our data to be more challenging than existing driving image recognition benchmarks, as it covers more realistic driving scenarios and captures more of the "long-tail" of appearance variation and pose configuration of categories of interest in diverse environmental domains. The major contributions of our paper are: a robust labeling system that is efficient and extensible, as well as a comprehensive diverse 100K driving video dataset that can serve as an evaluation benchmark for computer vision research for autonomous driving.

2 Related Works

Labeling tools have played an important role in generating annotations for supervised learning in computer vision [18,14,20,3]. Russell et al. [18] introduced a web-based

labeling tool, LabelMe, which is used to draw fine-grained polygons around relevant objects. While LabelMe only supports region annotation, our tool also supports different types of annotations with features to improve labeling efficiency. Lin et al. [14] broke instance segmentation into several steps to speed up the labeling process. Similarly, Vondrick et al. [20] delivered their interactive algorithm in an open source tool for annotating bounding boxes. These tools have been very useful in constructing datasets and have accelerated the progress of computer vision and deep learning in industry and academia, however, they all lack extensiveness and an interface for consolidation of annotations. The operations, such as drawing polygons, of the existing tools are primitive, which limits their efficiency. Our tool supports more operations, such as Bzier curve and boundary sharing, which make labeling more productive. Although some recent algorithms such as Poly-RNN [6] can also help generate complicated annotations, we focus on getting the accurate manual labels with minimal algorithm bias. Similar to [3], our system provides an administration interface to monitor labeling quality in real-time. Unlike most existing tools, which only support one type of annotation (e.g bounding boxes, segmentation, etc.), our annotation system delivers different types of annotation in one consistent pipeline.

Visual datasets are necessary for numerous recognition tasks in computer vision. Especially with the advent of deep learning methods, large scale visual datasets, such as [8,24,26], are essential for learning high-level image representations. They are general-purpose and include millions of images with image-level categorical labels. These large datasets with image-level labels are useful in learning representations for image recognition, but most of the complex visual understanding tasks in the real world require more fine-grained recognition such as object localization and segmentation [10]. Our proposed dataset provides these multi-granularity annotations for more in-depth visual reasoning. In addition, we provide these annotations in the context of videos, which provides an additional dimension of visual information. Although large video datasets exist, such as [5,1,19], they usually are restricted to image-level labels.

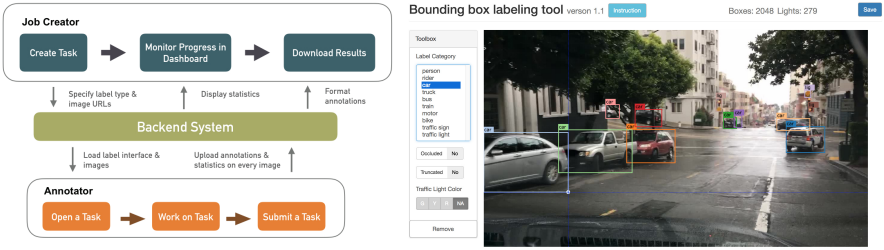
Driving datasets have received increasing attention in the recent years, due to the popularity of autonomous vehicle technology. The goal is to understand the challenge of computer vision systems in the context of self-driving. Some of the datasets focus on particular objects such as pedestrians [9,25]. Cityscapes [7] provides instance-level semantic segmentation on sampled frames of videos collected by their own vehicle. RobotCar [15] and KITTI [12] also provide data of multiple sources such as LiDAR scanned points. Because it is very difficult to collect data that covers a broad range of time and location, the data diversity across these datasets is limited. In order for a vehicle perception system to be robust, it needs to learn from a variety of road conditions from numerous cities. Our data was collected from the same original source as the videos in [22], however, the primary contribution of our paper is the single-frame annotation and annotation tooling, not the video sequences (and not an end-to-end driving model) reported in their paper. Mapillary Vistas [16] provides fine-grained annotations for user uploaded data, which is much more diverse with respect to location. However, these images are one-off frames that are not placed in the context of videos that contain temporal structure. Like Vistas, our data is crowdsourced, however, our dataset is

collected solely from drivers, with each annotated image corresponding to a video sequence, which enables some interesting applications for modeling temporal dynamics.

3 Labeling System

Our goal is to design a versatile and scalable annotation tooling that is suitable for all kinds of annotations needed in a driving database, such as bounding box, semantic instance segmentation, and lane detection. Many open source annotation tools are targeted to one specific task, such as single object classification or vehicle/pedestrian detection, however, no existing open source tool is available to support various types of annotations for such a large driving database. Also, in order to have an extensive collection of images annotated with maximum efficiency, the annotation work must be easily accessible to workers, the annotation progress needs to be monitorable, and concurrent annotation sessions need to be supported.

To satisfy these requirements, we design an efficient labeling system that is sufficient for the annotation of a large driving video database. First of all, the system is adaptable so that it would allow the practice of various kinds of annotation work such as bounding box and region annotation. Secondly, the system aims to make the annotation process as seamless and user-friendly as possible. Besides, we choose to implement the annotation system as a web-based tool so that it can be simply accessed through a web browser without installation. Figure 2 (a) shows an overview of our system. In this section, we will describe the details and evaluate the efficiency of our annotation tool.



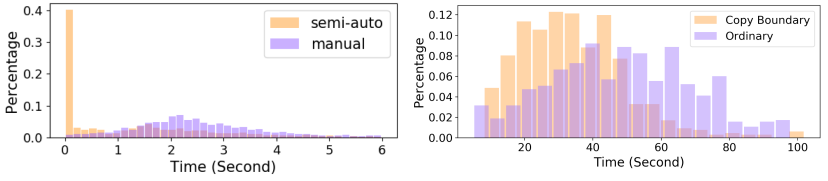
(a) Labeling tool workflow diagram. (b) Bounding box labeling tool.

Fig. 2: Back-end and front-end of our labeling system

3.1 Box Annotation

An interface example is shown in Fig. 2 (b). As a novel feature of our tool, suggestions on the category of a bounding box can be provided to annotators by using the outputs from an object detection model. After obtaining the annotations of the full 55k video clips, we trained a Fast-RCNN object detection model with IoU of 0.6. We set up the system by uploading the model outputs as initial annotations and ask annotators

to complete the annotation by adjusting or adding new bounding boxes. An experiment was conducted to evaluate the efficiency of the semi-automatic system, which integrates outputs from object detection models with the manual system from only human inputs. Annotators were asked to draw bounding boxes around all objects of 10 given categories in 2,000 images using the two different systems, then record the time spent on operating each bounding box. As shown in Fig. 3(a), the object detector is able to label 40% of bounding boxes at a minimal cost. On average, the time of drawing and adjusting each bounding box is reduced by 60%. Our study shows that our proposed semi-automatic system outperforms the manual process. To provide ground truth for future study of semi-automatic labeling systems, all the labeled provided in this paper are labeled manually.



(a) (b)

Fig. 3: (a) Distribution of time in seconds spent on drawing a new bounding box and adjusting a bounding box suggested by the object detection model. The histogram illustrates the percentage of bounding boxes drawn in time represented on the horizontal axis. (b) Distribution of time in seconds spent on drawing a polygon one by one, and using copying boundaries. By using this technique, the time is reduced significantly.

3.2 Region Annotation

The region annotation interface is used to annotate region or boundary information, such as semantic segmentation, drivable area, and lane markings. To support fine and quick annotations, techniques similar to [18,20,3], such as a local magnifier, zoom in/out, and keyboard shortcuts, are implemented. To further facilitate the accuracy and effectiveness of annotation processes, we also introduce several innovative features, which outperform previous methods in providing finer and quicker annotation tooling.

Bezier curve. To fit curved boundaries, we provide the option to draw parametric Bezier curves. Annotators can easily change any line segment to a cubic Bezier curve. By adjusting the shape of the curves, annotators could fit labels to objects with much higher accuracy and efficiency. As shown in Fig. 4, Bezier curve has apparent smoothness, compared with many dispersed clicks on a curved boundary.

Copy shared boundaries. In tasks such as segmentation annotations, there are usually many objects to be labeled in an image. As shown in Fig 5, objects share boundaries with their neighbors. To avoid drawing the same boundaries repeatedly, we introduce

(a) Polygon with dispersed clicks. (b) Polygon using Beizer curves.

Fig. 4: Compared to dispersed clicks in (a), Beizer curve in (b) provides us with more smoothness and less operations. The vertices connected by dashed lines are control points.

the function to automatically duplicate shared boundaries and make it possible for a polygon to share boundaries with its neighbors.

Fig. 5: Polygons with shared boundaries.

For example, when drawing a polygon A, if A's boundary partly overlaps with polygon B, which has already been drawn, the annotator can let the system automatically generate the shared boundary by clicking on two desired endpoints. In addition, adequate visual feedback is also implemented to smooth out the annotation process. The copy shared boundary feature not only provides unique and cleaner boundaries between adjacent objects, but also reduces the annotation time significantly. According to the our user study, in which annotators were asked to label 20 images with 842 polygons, the time to draw a polygon was reduced by 36% on average when using this technique as shown in Fig. 3(b).

3.3 Extensibility

Vision tasks are mostly about regrouping pixels on the images and assigning semantic meanings to them. For example, when only the object location and extent are needed, bounding boxes are easier to annotate and recognize. Therefore, we provide operations that can be used for labeling different tasks and we provide that options for the system user to configure the semantics of the targeted regions. We study this extensibility by

labeling large-scale real-world data with distinct types of annotations, which will be discussed in the following section.

4 Video Database

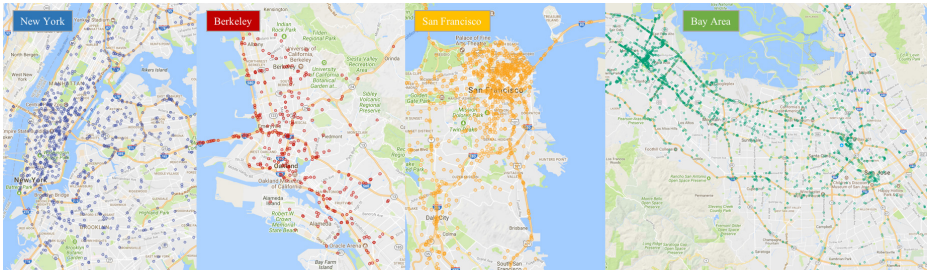
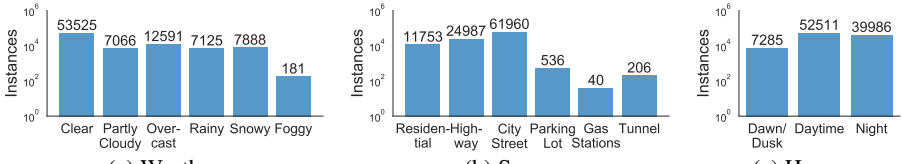


Fig. 6: Geographical distribution of sample data in four major regions (1000 samples in each region). Each dot represents the starting location of every video clip.

Powered by our annotation tool, we aim to provide a large-scale diverse driving video dataset with rich labels that reflects the challenges of street-scene understanding. To achieve good diversity, we obtain our videos in a crowd-sourcing manner. The videos are from tens of thousands of rides of normal drivers. They contain not only high-resolution (720p) and high-framerate (30fps) images, but also GPS/IMU information to record the trajectories. In total, we have 100K driving videos collected from more than 50K rides, covering New York, San Francisco Bay Area, and other regions as shown in Figure 6. Each video is 40-second long. The dataset contains diverse scene scenarios such as city streets, residential areas, and highways. Also, the videos were recorded in diverse weather conditions (sunny/rainy/snowy) and different time of day (daytime/nighttime/dusk/dawn). The frame at the 10^{th} second in each video is extracted and annotated. Examples are shown in Fig 1. In the rest of this section, we will discuss different types of annotations our annotation system provides.

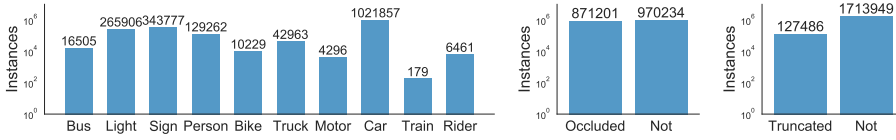
4.1 Image Tagging



(a) Weather (b) Scene (c) Hours

Fig. 7: Distribution of images in weather, scene, and day hours categories

We have collected image-level annotation on six weather conditions, six scene types, and three distinct times of day, for each image. As shown in Fig. 7, the videos contain large portions of extreme weather conditions - such as snow and rain. They also include a diverse number of different scenes across the world. Notably, our dataset contains approximately an equal number of day-time and night-time videos. Such diversity enables us to study domain transfer and generalize our object detection model well on new test sets, a point which will be elaborated further in section 6.1.



(a) Number of instances in each category. (b) Occlusion (c) Truncation

Fig. 8: Number of instances in our object categories.

4.2 Object Detection

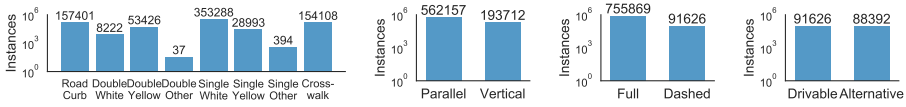
For each video clip from the 100K database, we provide bounding box annotations for the following 10 categories as shown in Figure 8a. We also provide details about whether a given object is "occluded" or "truncated" as shown in Fig. 8b and Fig. 8c. Furthermore, the light colors of all traffic lights are identified. The 100K image annotation file contains a list of labeled objects: each object includes source image URL, a category label, its size (starting coordinate, ending coordinate, width, and height), its attributes of truncation, occlusion, and traffic light color. On average, there are 9.7 cars and 1.2 persons in each image. Because our dataset has many different scene types and not all of them are crowded, it has fewer persons per image than most of the other datasets; however, our dataset do have a much larger total number of unique persons, as shown in Table 1. We also observe the long tail properties, as there are almost one million cars, but only more than one hundred trains in the dataset.

	Caltech [9]	KITTI [12]	City [25]	Ours
# persons	1,273	6,336	19,654	86,047
# per image	1.4	0.8	7.0	1.2

Table 1: Comparisons on number of pedestrians with other datasets. The statistics is only based on the training set in each dataset. Our dataset has more examples of pedestrians, but because our dataset contains non-city scenes such as highway, the number of person per image is lower than Cityscapes.

4.3 Lane

The lane marking detection is critical for vision-based vehicle localization and trajectory planing. Available datasets are often limited in scale and diversity. For example,

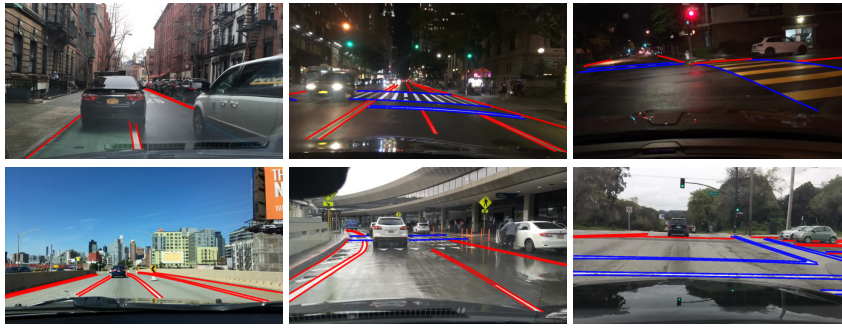


(a) Lane category (b) Lane direction (c) Lane continuity (d) Drivable areas

Fig. 9: Distribution of different types of lanes and drivable areas.

related works such as the Caltech Lanes Dataset [2] only contains 1,224 images; and the Road Marking Dataset [21] only contains 1,443 images labeled in 11 classes of lane markings. The most recent work, VPGNet [13] consists of about 20,000 images taken during three weeks of driving in Seoul, which are labeled in 17 classes.

As shown in Fig 9, in our driving dataset, the lane markings are annotated in 8 main categories (road curb, crosswalk, double white, double yellow, double other color, single white, single yellow, single other color), with attributes of continuity (full or dashed) and direction (parallel or vertical). Compared to other datasets, our lane marking annotations cover more classes.



(a) (b) (c)

Fig. 10: Examples of lane marking annotations. Red lanes are vertical and blue lanes are parallel. (a) We label all the visible lane boundaries. (b) Not all marking edges are lanes for vehicles to follow, such as pedestrian crossing. (c) Parallel lanes can also be along the current driving direction.

Moreover, there is methodology behind our choices for lane annotation. Usually, the continuity of a lane marking is essential for making a "driving-across" decision, so we labeled it independently as an important attribute. Similarly, the direction of a lane marking is also significant for autonomous driving. For example, if a lane marking is parallel to the passing car, it may serve to guide cars and separate lanes; if it is vertical, it can be treated as a sign of deceleration or stop. The distribution of the number of annotations in varied driving scenes are shown in Fig 9a, Fig 9b, and Fig 9c.

4.4 Drivable Area

Lanes alone are not sufficient to decide road affordability for driving. Although most of the time, the vehicle should stay between the lanes, it is common that no clear lane

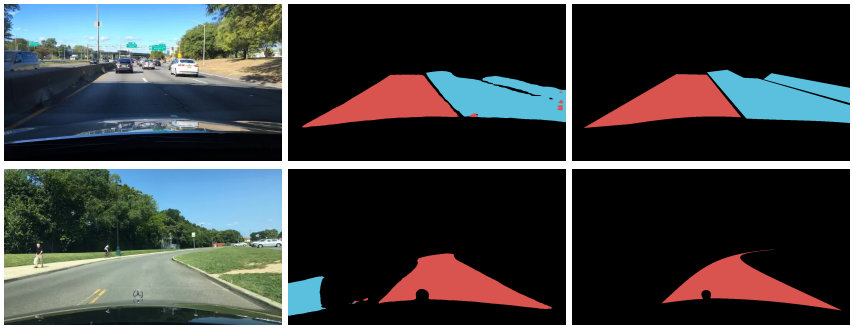
	Training	Total	Sequences	Weather	Time	Attributes
Caltech Lanes Dataset [2]	-	1,224	4	1	1	2
Road Marking Dataset [21]	-	1,443	29	2	3	10
KITTI-ROAD [11]	289	579	-	1	1	2
VPGNet [13]	14,783	21,097	-	4	2	17
Ours	70,000	100,000	100,000	6	3	11

Table 2: Comparisons with other lane marking datasets. Our annotations are significantly richer and are more diverse.

marking exists. In addition, the road area is shared with all other vehicles. A lane can not be driven on if occupied. All these conditions beyond lane markings direct our driving decisions, and are relevant for designing autonomous driving algorithms.

Fig. 11: Examples of drivable areas. Red regions are directly drivable and the blue ones are alternative. Although drivable areas can be confined within lane markings, they are also related to locations of other vehicles, as shown in the first row. The second row shows that some areas are perceptively drivable, even though no visible lane marking exists.

To support the study of drivable areas, we propose a new methodology beyond road segmentation. The drivable area is divided into two different categories: “directly drivable area” and “alternatively drivable area”. In our dataset, the ”directly drivable area” defines the area that the driver is currently driving on – it is also the region where the driver has priority over other cars or the “right of the way”. In contrast, “alternatively drivable are” is a lane the driver is currently not driving on, but could do so – via changing lanes. Although the directly and alternatively drivable areas are visually indistinguishable, they are functionally different, and requires potential algorithms to recognize blocking objects and scene context. The distribution of drivable region annotations is shown in Fig. 9d. Some examples are shown in Fig. 11. In align with our understanding, on highway or city street, where traffic is closely regulated, drivable areas are mostly within lanes and they do not with the vehicles or objects on the road. However, in residential areas, the lanes are sparse. Our annotators can judge what is drivable based on the surroundings.



(a) Image (b) DRN (c) Ground truth

Fig. 12: Drivable area prediction by segmentation. The segmentation predicts the drivable area with lanes well, as shown in the top row. Also, we find that the segmentation model learns to interpolate in areas that has no lane markings.

Drivable area detection is a new task. We show a simple baseline method on the task here. First, the drivable area detection is converted to 3-way segmentation task (background, directly, and alternatively drivable) by ignoring the region ID. Then, we train DRN-D-22 model [23] on the 70,000 training images. Some visual results on the validation set are shown in Fig. 12. We find that after learning from the large-scale image dataset, the model learns to split the road according to the lanes and extrapolate the drivable area to unmarked space. The mIoU for directly and alternatively drivable areas is 77.6% and 59.7%, respectively. However, the same model achieves 94.4% IoU on road segmentation, as shown in Fig. 4, which indicates that techniques beyond segmentation may be required to solve the drivable area problem.

4.5 Semantic Instance Segmentation

We also provide fine-grained, pixel-level annotations for images from each of the 5,683 video clips randomly sampled from the whole dataset. Each pixel is given a label and a corresponding identifier denoting the instance number of that object label in the image. Since many classes (e.g sky) are not amenable to being split into instances, only a small subset of class labels are assigned instance identifiers. The entire label set consists of 40 object classes that are chosen to capture the diversity of objects in road scenes as well as maximizing the number of labeled pixels in each image. In addition to having a large number of labels, our dataset exceeds previous efforts in terms of scene diversity and complexity, a point that will be discussed further in Section 6.2. The whole set is split into 3 parts for training (3,683 images), validation (500 images), and testing (1,500 images), respectively.

In Fig. 13, we provide a distribution of the label set with respect to the number of instances observed across the segmentation dataset. There is good coverage on rare object categories (e.g. trailer, train) and large number of instances of common traffic objects (e.g. car, person, etc.). We observe long-tail effects even on our dataset. There are almost 60,000 car instances, but only tens for trailer and train, and several hundreds for rider and motorcycle.

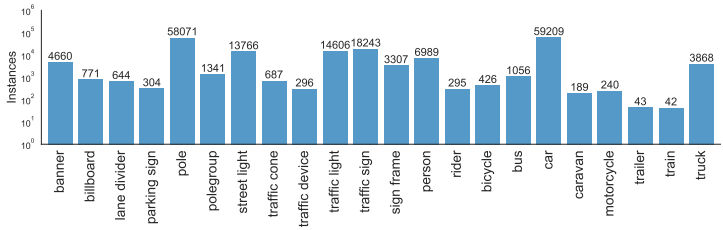


Fig. 13: Distribution of classes in semantic instance segmentation. It presents a long-tail effect with more than 10 cars and poles per image, but only tens of trains in the whole dataset.

5 Diversity

One of the distinct features of our data is diversity, besides video and scale. Given the labeled dataset, we want to study new challenges that the diversity brings to existing algorithms and how our data complements existing datasets. We conduct two sets of experiments: object detection and semantic segmentation. In object detection experiments, we study the different domains within our dataset. While in semantic segmentation, we investigate the domains between our data and Cityscapes.

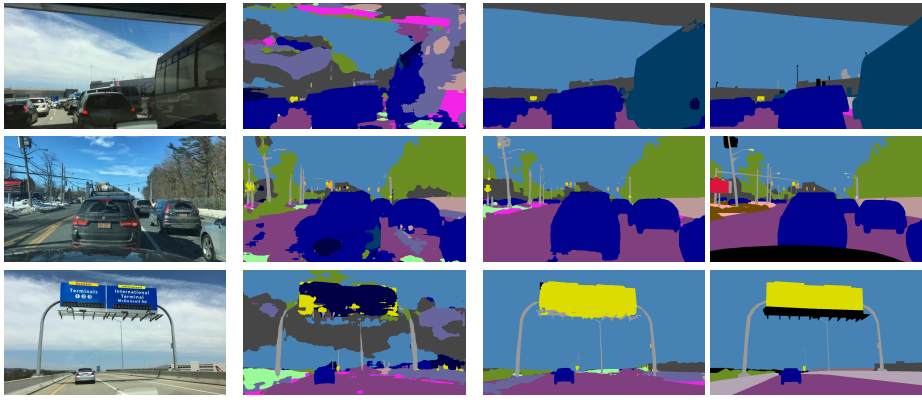
5.1 Object Detection

Train	Test	bike	bus	car	motor	person	rider	light	sign	train	truck	mAP
clear		20.5	37.3	69.6	9.9	45.4	22.3	39.7	50.7	0.0	44.7	34.0
daytime	out-domain	18.0	26.5	56.7	5.9	37.8	16.9	21.7	41.5	0.0	34.2	25.9
city		21.1	25.2	68.8	28.9	41.5	22.3	42.1	56.4	0.0	39.2	34.5
clear		26.9	42.6	71.2	18.4	41.9	22.6	41.2	56.4	0.0	44.9	36.6
daytime	in-domain	28.5	41.0	70.1	18.9	46.8	30.0	38.7	45.8	0.0	45.9	36.6
city		38.0	47.9	72.9	19.3	52.8	34.8	47.8	55.6	0.0	50.4	42.0

Table 3: Domain Discrepancy Experiments with Faster-RCNN. We take the images from one domain such as daytime in training set and report testing results on the same domain or the opposite domain such as non-daytime. Although there is not much domain difference for the weather, different time of the day and different scene types have large performance discrepancies.

Compared to existing popular driving datasets like Cityscapes [7], or Camvid [4], which do not include different scene types and conditions, our dataset has a diversity advantage because it contains information on weather conditions, daytime, and scene location. We conduct object detection experiment to investigate the domain difference and its impact on existing algorithms. The widely-used Faster-RCNN [17] algorithm is used to train models based on weather, scene type, and time of the day. Clear weather, city street and daytime are chosen as training domains, which have similar number of

images (around 36,000) in the training set. Then, models trained on these three subsets are tested on the same domains or opposite domains in the testing set. The quantitative results are shown in Table 3. We find that the in-domain and out-domain differences for model training on clear weather is insignificant. However, the model trained on daytime performs poorly on the other time of the day, mainly nighttime, which indicates that lighting is still an important factor for model transfer. Also, the model trained on city street images also performs poorly out-of-domain, mainly highway and residential area, which confirms that context change is important for domain transfer. Our dataset would become a useful testing bed for domain transfer solutions between datasets of real images.



(a) Our Image (b) Trained on Cityscapes (c) Trained on Ours (d) Ground Truth

Fig. 14: Visual comparisons of the same model trained on different datasets. We find that there is a dramatic domain shift between Cityscapes and our new dataset. Especially, because of infrastructure difference, the model trained on Cityscapes is confused by some simple categories such as sky and traffic signs.

5.2 Semantic Segmentation

To understand the difference between our new datasets and existing driving datasets, we also compare the models trained on Cityscapes and ours. Cityscapes data is collected from German cities, while our data is mainly from the US. We convert our semantic segmentation to label maps with training indices specified in Cityscapes. For comparison, we test the models in 4 different settings based on sources of training and testing sets as listed in the second and third columns of Table 4. Validation sets are used in both datasets for this ablation study. We take dilated residual networks [23] which perform well on Cityscapes. In addition, we use their pre-trained model on Cityscapes, and train the models with the same hyperparameters on our dataset. Because our images are smaller, the crop size is changed to 688. Quantitative results are shown in Table 4.

We observe that there is a dramatic domain shift between the two datasets for semantic segmentation models. The models perform much worse when tested on a different dataset. The performance drops significantly for objects that move on the road, such as riders and motorcycles. However, models trained on Cityscapes that work better

also work better on our data. Surprisingly, when we train the models on our dataset and test them on Cityscapes, not all the categories have lower accuracy than those trained in domain, such as bicycle and truck, although the overall performance is worse. This suggests that even for the domain of other datasets, our new dataset is complementary, which augments existing datasets.

	Train	Test	Road	Sidewalk	Building	Wall	Fence	Pole	Light	Sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	mean IoU
DRN-D-22	City	City	97.2	79.9	90.2	38.0	46.6	58.8	63.3	73.1	91.2	57.1	93.8	77.6	52.4	92.4	41.0	68.5	52.7	45.4	73.0	68.0
	City	Ours	60.0	27.7	55.9	3.4	17.3	31.8	31.6	34.6	76.2	19.3	77.0	42.6	6.4	62.2	10.1	9.8	0.0	9.0	9.7	30.8
	Ours	Ours	94.4	57.0	83.2	24.0	42.0	46.6	48.6	54.4	86.1	44.7	96.8	53.8	29.5	88.0	45.6	52.2	0.3	38.8	24.8	53.2
	Ours	City	89.4	52.7	80.0	14.1	9.7	43.7	31.4	32.5	24.4	86.2	62.6	57.8	13.2	24.8	48.2	12.3	12.8	1.3	80.6	40.9
DRN-D-38	City	City	97.7	82.2	91.4	45.1	51.2	61.7	67.3	75.4	91.8	58.4	94.1	79.8	57.9	93.5	53.9	73.2	52.1	54.1	74.8	71.4
	City	Ours	79.9	42.0	63.6	6.7	20.1	36.8	35.7	36.2	78.0	25.3	82.3	48.8	13.9	71.2	18.9	15.9	1.0	22.3	22.3	37.9
	Ours	Ours	95.3	61.6	84.9	25.1	45.0	49.1	51.7	57.8	86.8	47.8	97.0	60.9	34.6	89.7	51.6	56.9	0.0	32.2	21.7	55.2
	Ours	City	91.9	59.8	83.8	19.2	15.0	47.5	36.1	41.0	29.1	86.8	76.9	62.8	19.1	28.4	52.9	21.7	14.3	1.1	85.5	45.9

Table 4: Domain discrepency properties of semantic segmentation models. We train the Dilated Residual Networks [23] and evaluate on both Cityscapes and our data. Quantitatively, we observe a dramatic domain shift between the two datasets. We also find that models trained on our dataset perform well on Cityscapes in some categories such as bicycle and truck.

As shown in Figure 14, we implement visualizations on some segmentation examples produced by DRN-D-38. They also reveal some interesting properties of different domains. Probably because of the infrastructure differences between Germany and the US, the models trained on Cityscapes confuse some big structures in an unreasonable way, such as segmenting the sky as building as shown in the third row in Figure 14. The model is also confused by the US highway traffic sign. However, the same model trained on our dataset does not suffer these problems. Also, the model of Cityscapes may over-fit the hood of the data collecting vehicle and produces erroneous segmentation for the lower part of the images.

6 Conclusion

Our contributions from this paper are two-fold: 1) a robust video annotation system as well as 2) a comprehensive large-scale driving dataset with extensive annotations. First, the annotation system is an improvement over existing solutions in terms of efficiency and extensibility. Our annotation system incorporates different kinds of labeling heuristics to improve productivity, and can be extended to different types of image annotation. With this production-ready annotation system, we are able to label a driving video dataset that is larger and more diverse than existing datasets. This dataset comes with comprehensive annotations that are necessary for a complete driving system. Moreover, experiments show that this new dataset is more challenging and more comprehensive than existing ones, and can serve as a good benchmark for domain adaption due to its diversity. This will serve to help the research community with understanding on how different scenarios affect existing algorithms' performance.

References

1. Abu-El-Haija, S., Kothari, N., Lee, J., Natsev, P., Toderici, G., Varadarajan, B., Vijayanarasimhan, S.: Youtube-8m: A large-scale video classification benchmark. arXiv preprint arXiv:1609.08675 (2016)
2. Aly, M.: Real time detection of lane markers in urban streets. In: Intelligent Vehicles Symposium. pp. 7–12 (2008)
3. Bell, S., Upchurch, P., Snavely, N., Bala, K.: Opensurfaces: A richly annotated catalog of surface appearance. ACM Transactions on Graphics (TOG) 32(4), 111 (2013)
4. Brostow, G.J., Shotton, J., Fauqueur, J., Cipolla, R.: Segmentation and recognition using structure from motion point clouds. In: European conference on computer vision. pp. 44–57. Springer (2008)
5. Caba Heilbron, F., Escorcia, V., Ghanem, B., Carlos Niebles, J.: Activitynet: A large-scale video benchmark for human activity understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 961–970 (2015)
6. Castrejón, L., Kundu, K., Urtasun, R., Fidler, S.: Annotating object instances with a polygon-rnn. In: CVPR. vol. 1, p. 2 (2017)
7. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3213–3223 (2016)
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 248–255. IEEE (2009)
9. Dollár, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: A benchmark. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 304–311. IEEE (2009)
10. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision 88(2), 303–338 (2010)
11. Fritsch, J., Kuhnl, T., Geiger, A.: A new performance measure and evaluation benchmark for road detection algorithms. In: Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on. pp. 1693–1700. IEEE (2013)
12. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The International Journal of Robotics Research 32(11), 1231–1237 (2013)
13. Lee, S., Kim, J., Yoon, J.S., Shin, S., Bailo, O., Kim, N., Lee, T.H., Hong, H.S., Han, S.H., Kweon, I.S.: VPGNet: Vanishing point guided network for lane and road marking detection and recognition. In: Computer Vision (ICCV), 2017 IEEE International Conference on. pp. 1965–1973. IEEE (2017)
14. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
15. Maddern, W., Pascoe, G., Linegar, C., Newman, P.: 1 year, 1000 km: The oxford robotcar dataset. IJ Robotics Res. 36(1), 3–15 (2017)
16. Neuhold, G., Ollmann, T., Bulò, S.R., Kontschieder, P.: The mapillary vistas dataset for semantic understanding of street scenes. In: International Conference on Computer Vision (ICCV) (2017)
17. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)

18. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: Labelme: a database and web-based tool for image annotation. International journal of computer vision 77(1), 157–173 (2008)
19. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
20. Vondrick, C., Patterson, D., Ramanan, D.: Efficiently scaling up crowdsourced video annotation. International Journal of Computer Vision 101(1), 184–204 (2013)
21. Wu, T., Ranganathan, A.: A practical system for road marking detection and recognition. In: Intelligent Vehicles Symposium. pp. 25–30 (2012)
22. Xu, H., Gao, Y., Yu, F., Darrell, T.: End-to-end learning of driving models from large-scale video datasets. arXiv preprint (2017)
23. Yu, F., Koltun, V., Funkhouser, T.: Dilated residual networks. In: Computer Vision and Pattern Recognition (CVPR) (2017)
24. Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., Xiao, J.: LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop. arXiv preprint arXiv:1506.03365 (2015)
25. Zhang, S., Benenson, R., Schiele, B.: Citypersons: A diverse dataset for pedestrian detection. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
26. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: Advances in neural information processing systems. pp. 487–495 (2014)