

# A Fine-Grained Vehicle Detection (FGVD) Dataset for Unconstrained Roads\*

Prafful Kumar Khoba  
IIT Delhi, India  
qiz228274@iitd.ac.in

Chirag Parikh  
IIIT Hyderabad, India  
chirag.parikh@students.iiit.ac.in

Rohit Saluja  
IIIT Hyderabad, India  
rohit.saluja@research.iiit.ac.in

Ravi Kiran Sarvadevabhatla  
IIIT Hyderabad, India  
ravi.kiran@iiit.ac.in

C.V. Jawahar  
IIIT Hyderabad, India  
jawahar@iiit.ac.in

## ABSTRACT

The previous fine-grained datasets mainly focus on classification and are often captured in a controlled setup, with the camera focusing on the objects. We introduce the first Fine-Grained Vehicle Detection (FGVD) dataset in the wild, captured from a moving camera mounted on a car. It contains 5502 scene images with 210 unique fine-grained labels of multiple vehicle types organized in a three-level hierarchy. While previous classification datasets also include makes for different kinds of cars, the FGVD dataset introduces new class labels for categorizing two-wheelers, autorickshaws, and trucks. The FGVD dataset is challenging as it has vehicles in complex traffic scenarios with intra-class and inter-class variations in types, scale, pose, occlusion, and lighting conditions. The current object detectors like yolov5 and faster RCNN perform poorly on our dataset due to a lack of hierarchical modeling. Along with providing baseline results for existing object detectors on FGVD Dataset, we also present the results of a combination of an existing detector and the recent Hierarchical Residual Network (HRN) classifier for the FGVD task. Finally, we show that FGVD vehicle images are the most challenging to classify among the fine-grained datasets. [GitHub]

## CCS CONCEPTS

• Computing methodologies → Object detection.

## KEYWORDS

Fine-grained, detection, dataset, unconstrained roads, gradcam.

### ACM Reference Format:

Prafful Kumar Khoba, Chirag Parikh, Rohit Saluja, Ravi Kiran Sarvadevabhatla, and C.V. Jawahar. 2022. A Fine-Grained Vehicle Detection (FGVD) Dataset for Unconstrained Roads. In *Proceedings of the Thirteenth Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP'22)*, December 8–10, 2022, Gandhinagar, India, Soma Biswas, Shanmuganathan Raman, and Amit K Roy-Chowdhury (Eds.). ACM, New York, NY, USA, Article 83, 9 pages. <https://doi.org/10.1145/3571600.3571626>

\*Produces the permission block, and copyright information

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](https://permissions.acm.org).

ICVGIP'22, December 8–10, 2022, Gandhinagar, India

© 2022 Copyright held by the owner/author(s). Publication rights licensed to ACM. ACM ISBN 978-1-4503-9822-0/22/12. <https://doi.org/10.1145/3571600.3571626>

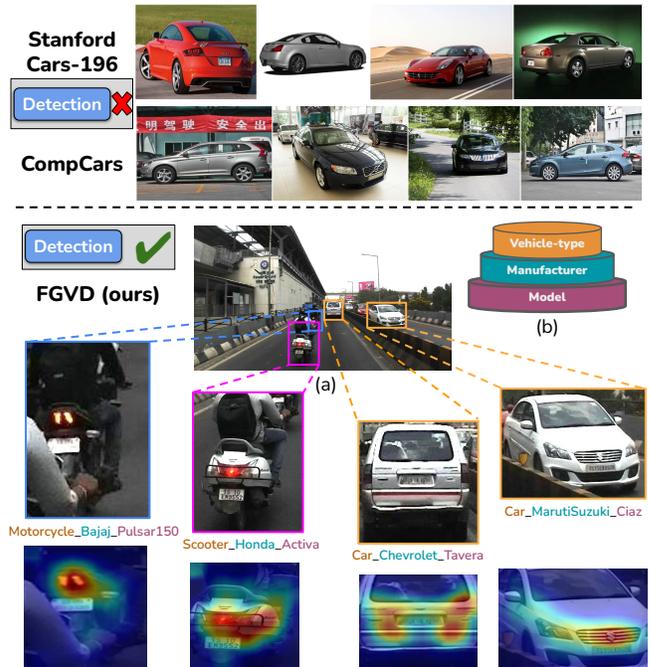


Figure 1: Fine-grained datasets samples. Top: Previous datasets focus only on the classification of cars on vehicle-centric images. Middle: The proposed FGVD dataset enables fine-grained (multi) vehicle detection on unconstrained road scenes captured from vehicle-mounted cameras. Bottom left GradCAM++ [1] visualizations on predicted crops show that the model focuses on the backlight and blinker at the motorcycle's top and scooter's bottom. For Tavera, the design on the left and right of license plates, and for Ciaz, the radiator and headlight regions are highlighted classification features.

## 1 INTRODUCTION

Intelligent traffic monitoring systems are of utmost need in big cities for public security, planning, and surveillance. For the tasks like vehicle re-identification and robust detection (e.g., when a vehicle occludes another vehicle that is similar in appearance), the detectors used in the surveillance systems should finely classify the vehicle type, manufacturer, and model of the on-road vehicles. Conventionally, detection models like YOLO [12] and Faster RCNN [14] are trained to classify vehicles based on coarse categories of on-road datasets like BDD and IDD [18, 22]. A coarse class can

**Table 1: Related Fine-grained datasets. All the previous road scene works include fine-grained classification only on car types. In contrast, FGVD has fine-grained detection labels for seven different types of vehicles (refer Fig. 2 and Table 2).**

Dataset	Source	# Levels	# Vehicle-type	Classification	Detection
BoxCars116k [15]	CCTV	3	1	√	×
CompCars [21]	Web/CCTV	3	1	√	×
THS-10 [10]	CCTV	2	1	√	×
Stanford Cars-196 [7]	Web	2	1	√	× <sup>ψ</sup>
<b>FGVD (ours)</b>	<b>Dashcam</b>	<b>3</b>	<b>6</b>	√	√

<sup>ψ</sup> Stanford Car-196 is a part-based fine-grained car recognition dataset containing bounding boxes for cars' parts.

contain multiple sub-classes with minute variations, referred to as the fine-grained classes. The localization of such sub-class vehicles based on their granularity in the design is known as Fine-Grained Vehicle Detection (FGVD). The FGVD models and datasets can enable robust vehicle re-identification and detection in highly dense and occluded traffic scenarios. Therefore, we propose a novel FGVD dataset with multiple hierarchy levels for the fine-grained labels. Fig. 1 depicts a sample scene image from the FGVD dataset and the corresponding labels. As shown in the figure, in addition to enabling the detection task, the FGVD dataset includes complex intra-class and inter-class variations in types, scales, and orientations compared to the previous fine-grained classification datasets. The dataset also contains challenging occlusion scenarios and lighting conditions (refer to Fig. 2). FGVD comprises three levels of hierarchy, i.e., vehicle type, manufacturer, and model, as shown at the bottom of the vehicles' Regions of Interest (ROIs) in Fig. 1 (a) and in Fig. 1 (b) with three different colors.

While some classification datasets like Stanford Cars-196 [7] and CompCars [21] also includes the makes for different kinds of cars (see Fig. 1 top), we introduce the complementary hierarchical labels for two-wheelers, autorickshaws, trucks, and buses (refer Sec. 3). The granularity of the FGVD dataset increases as we move from parent to child level (refer to Figs. 1 (b) and 4). For the hierarchical FGVD dataset, every level has its uniqueness. Firstly, different vehicles may look similar at the first level of granularity, e.g., motorcycles and scooters, both being the two-wheelers. However, as it can be inferred from Fig. 1, the overall design of the scooter is different from the motorcycle, e.g., scooters have a backlight at the bottom as compared to the top backlight of the motorcycle. The overall appearance of the two vehicles with the same parent may look even more similar. However, the minute subtle and local differences are present in the same subcategory. Also, some categories are not present in the earlier vehicle datasets, for example, scooter, autorickshaw, truck, and bus; hence, they must be added to a fine-grained dataset. Therefore, we introduce classes unique to the FGVD dataset to facilitate detailed research in fine-grained on-road scenarios. The main contributions of this work are as follows:

- A novel Fine-Grained Vehicle Detection (FGVD) dataset for on-road vehicles in dense and occluded traffic scenarios. To the best of our knowledge, no fine-grained detection dataset exists in the literature, and ours is the first dataset of its kind.
- We present the results of baseline detection and classification models on the proposed dataset. We also show the initial results of a combination of a detector and a recent hierarchical fine-grained classification model.

## 2 RELATED WORK

**Fine-grained Classification Datasets:** Early fine-grained classification datasets mainly focus on birds [19], flowers [11], dogs [6, 17], aircrafts [9]. The datasets like birds [19] cover appropriate occlusion examples, and the aircrafts dataset [9] has objects at low resolution showcasing sufficient complexity for the classification task. But the objects in these datasets are usually located around the image center, making it unfit for their detection in the wild. There are few public datasets for fine-grained vehicle classification with hierarchical labels of the car's make, model, submodel, and year of manufacture [10, 15, 21]. Jakub et al. [15] publish the BoxCars116k dataset with 3D bounding box annotations of 1, 16, 286 car images with 693 fine-grained classes. Yang et al. [21] released the CompCars dataset, which contains 44, 481 frontal view images of cars taken from surveillance cameras. It also contains web-nature images from different viewpoints of 136K vehicles classified into 600 categories. Najeeb et al. [10] released 4250 CCTV car images of 10 different models. Similar to this, the Stanford Cars dataset [7] includes 16, 185 vehicle images of 197 car types and their part labels to assist in fine-grained recognition tasks. All the above-mentioned recognition datasets contain only one car per image and lack the complexity required for detection in real-world traffic scenes. Our detection dataset includes diverse traffic scenarios observed in urban settings capturing large variations in scale, pose, occlusion, illumination, and density of vehicles. All the vehicle classification datasets mentioned above have fine-grained labels only for cars. In contrast, our dataset introduces these labels for four additional vehicles - motorcycles, scooters, autorickshaws, and trucks. Also, the proposed FGVD dataset contains images captured from dashboard cameras installed on top of surveillance vehicles instead of using static CCTV cameras, making its use economically viable and sustainable for road safety in any remote city location. The datasets like BDD100k [22], Waymo [16], and IDD [18] do focus on vehicle detection but lack fine-grained labels.

**Detection and Fine-grained Classification Models:** Recent detection models like Yolov5 [5] and Faster-RCNN [14] consider all object labels as independent from each other. So, they do not model the hierarchical relationship between the object's fine-grained labels. The difficulty of detecting fine-grained objects thus increases with deeper class definition as the number of samples per class becomes smaller, and the visual cues become more challenging. Chen et al. [2] recently proposed Label Relation Graphs Enhanced Hierarchical Residual Network (HRN), which gives state-of-the-art performance on fine-grained classification datasets. Their architecture exploits the parent-child correlation between labels by transferring

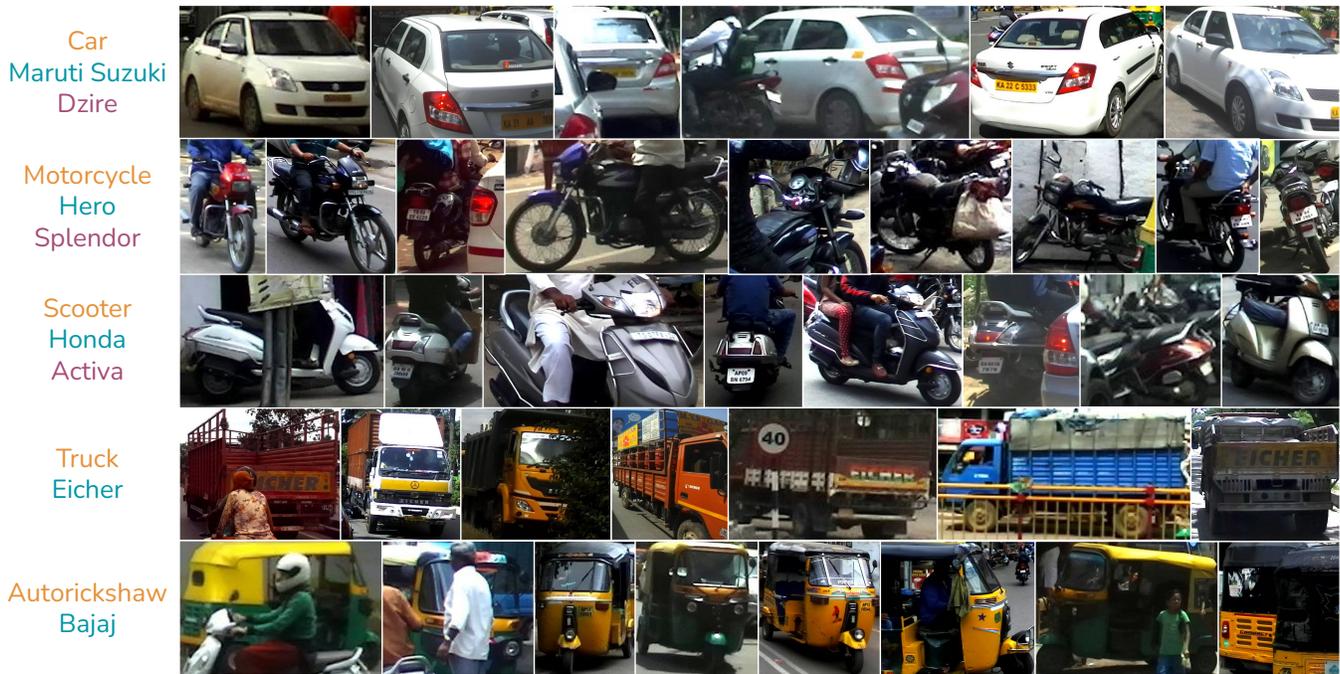


Figure 2: Sample images of different categories in FGVD exhibiting inter-class similarities (bottom two rows), multiple vehicle orientations (all rows), and frequent occlusions.



Figure 3: Annotation Strategy: (b) is the only confidently identifiable image frame where the label cue (logo at bottom-left of the car) is visible. As the frames are from common video sequence, the label is propagated to the vehicle instances in (a), (c) & (d).

the hierarchical knowledge through residual connections across feature levels, but it is not used for object detection. Therefore, we combined the object detection and classification models and obtained superior performance on fine-grained vehicle detection.

### 3 FGVD DATASET AND ANNOTATION

We use images with corresponding coarse labels, and bounding boxes from the IDD detection dataset [18]. The vehicles far from the camera are infeasible to annotate. Hence, we remove the bounding boxes with a height-to-width ratio lesser than the thresholds, which are different for different vehicle types. To consider the variability of vehicles in physical dimensions, we keep the threshold for the truck's bounding box ratio higher than that of the car, which in turn is higher than that of the bike. The thresholding process makes the

annotators' work easy, manageable, and quick. The FGVD dataset contains 5502 scene images containing around 24450 bounding boxes, with 217 (210 unique, and 7 repeated from higher levels) fine-grained labels in the third level.

#### 3.1 Annotation Process

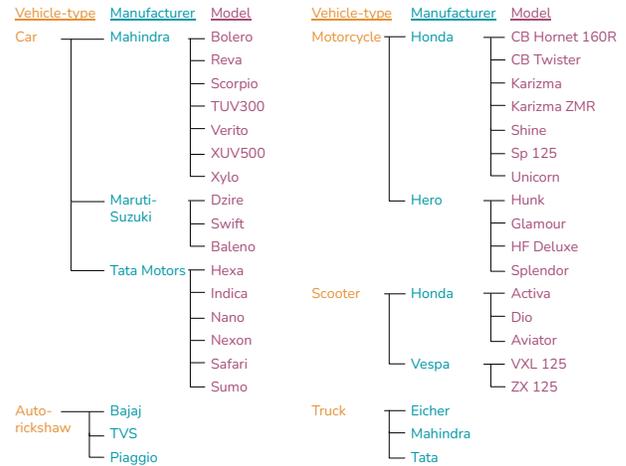
We select 5502 out of 16311 high-quality images from the IDD-Detection dataset based on occlusion, size of vehicle boxes, and traffic density. The annotation team consists of four highly skilled annotators and two expert reviewers for quality checks. Firstly, we train the annotators for the task by providing fine-grained labels for a few FGVD samples. Secondly, we provide the guidelines, template, and a list of objects to be annotated. After proper training, the annotators can recognize the popular vehicles in the scene. However,

Vehicle Type	Levels of Hierarchy	L-2 labels	L-3 labels
Car	3	22	112
Motorcycle	3	11	67
Scooter	3	9	23
Truck	2	7	7
Auto-rickshaw	2	6	6
Bus	2	2	2
<b>Total</b>	<b>3</b>	<b>57</b>	<b>217</b>

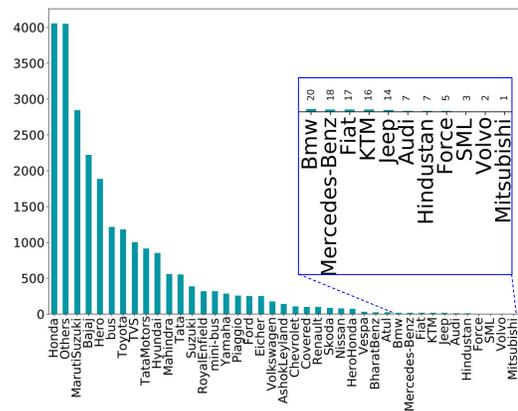
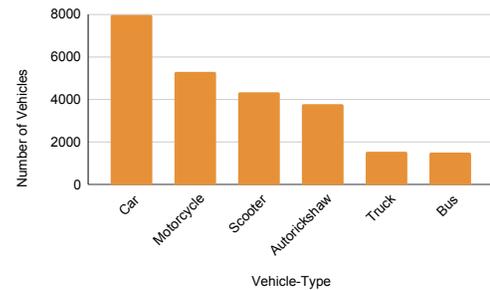
**Table 2: Levels of Hierarchy for different Vehicles in FGVD.**

if still, the vehicle is not recognizable, then they can use google lens or image search on the internet. For example, consider the scenario where the annotator can recognize the manufacturer by looking at the brand logo. Still, the model name is not visible due to occlusion, truncation, or any other complexity. In such scenarios, the annotator would search for similar vehicles on the manufacturer’s website. The essential part of looking at the image for classification is the vehicle’s overall design, design of its components (e.g., some scooters have petrol openers at the back), brand logo, and model name. While creating IDD-Detection dataset [18], many images are taken from the continuous video frames; therefore, images have a temporal connection. For instance, in Fig. 3 Tata Sumo is not confidently identifiable in the first frame due to truncation, but in the second frame, the brand label is visible, which gives the confidence to the annotator to label even when they are unable to recognize the vehicle’s design. Similarly, the annotators propagate the label in the next frame, in which there is a lot of occlusion from other vehicles, and it is comparatively hard to annotate if they do not connect the knowledge from different frames.

We divide the data creation process into two steps, i.e., the pilot phase and the takeoff phase. In the pilot phase, each annotator labels a small set of images containing less traffic density. We also train the annotators to label the vehicles from the dataset images, which are confidently identifiable but do not have a bounding box. In the case of cars, motorcycles, and scooters, we have an attribute called *new*. Whenever annotators encounter a new variant of any model in our dataset, they tickmark the *new* attribute in the checkbox. During the pilot phase, we note the average time to annotate one image, from which we estimate the number of days for creating the whole dataset. Performance in the pilot phase lets us choose the best two annotators as reviewers in the next stage. In the takeoff phase, many images contain high traffic density, which in turn causes massive occlusion (samples shown in Fig. 2). All the labels in the takeoff stage are reviewed. The images having labels with high confidence from the reviewers are selected to be part of the dataset. The remaining vehicle’s bounding boxes, for which any fine-grained levels are ambiguous, are labeled as “others”. If any material like a vehicle cover or a cloth covers any vehicle, they mark it as “covered”. The average time to annotate one image is one minute and half a minute to perform a quality check. We annotate all the scenes in the FGVD dataset using the Computer Vision Annotation Tool (CVAT)<sup>1</sup>



**Figure 4: Sample Hierarchy Tree of the FGVD dataset.**



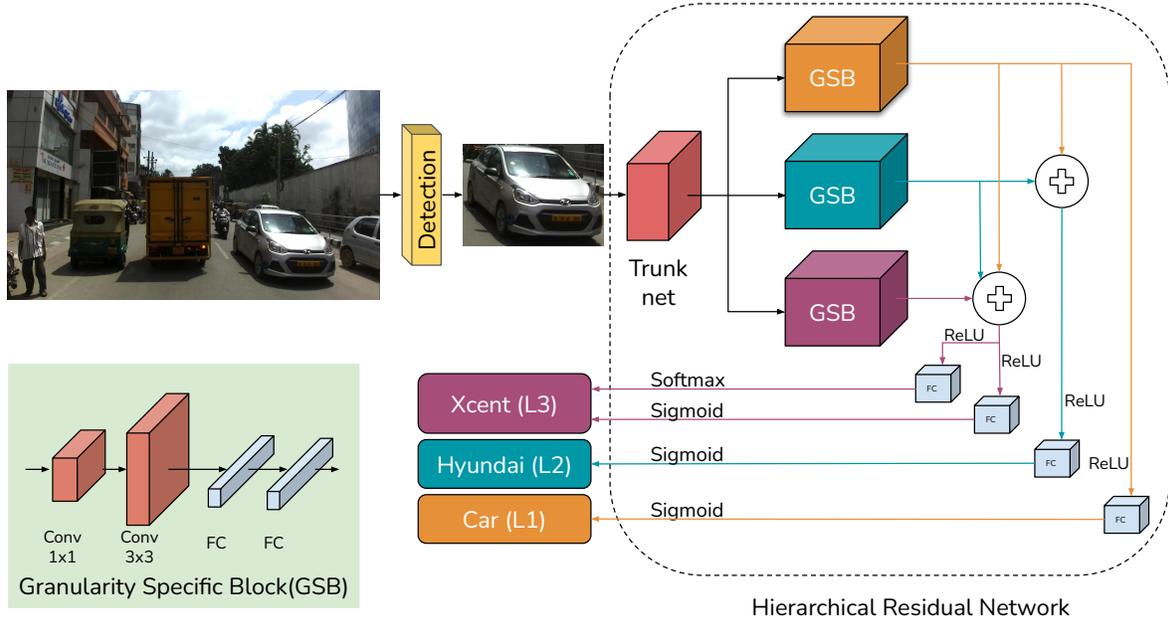
**Figure 5: Histograms for Level-1 (top) and Level-2 (bottom) labels in FGVD.**

### 3.2 Hierarchy and Long-tailed Distribution

As shown in Table 2, each vehicle type in the proposed Fine-Grained Vehicle Detection (FGVD) dataset has different hierarchical levels. We illustrate the sample vehicle images and hierarchy tree of the FGVD dataset in Figs. 2 and 4. As shown, the FGVD contains three different levels of hierarchy, which we detail below<sup>2</sup>:

<sup>1</sup><https://github.com/openvinotoolkit/cvat>

<sup>2</sup>For bus class, we do not follow this hierarchy. Small buses, known as “mini-bus,” and the “general-bus” are the child classes for the bus.



**Figure 6: Architecture for fine-grained vehicle detection. We use YOLO [5] for localization and Label Relation Graphs Enhanced HRN [2] for classification.**

- **Vehicle-type:** The highest coarse level labels of the vehicle come under the vehicle-type category. We consider it as level 1 of the hierarchy. Car, motorcycle, scooter, truck, auto-rickshaw, and bus are the six categories present in vehicle type.
- **Manufacturer:** The manufacturer level contains the primary producer of the vehicles. The manufacturer category has finer details than the vehicle type level. A producer may manufacture multiple kinds of vehicles. For example, Bajaj manufactures motorcycles as well as auto-rickshaw.
- **Model:** The model level is at the last group of the hierarchy. This level comprises highly fine-grained features that are unique for the variant. For example, a car’s design must be unique for each manufacturer.

As illustrated in Fig. 5, the annotation levels contain the common challenge of a class imbalance to different degrees.

## 4 METHODOLOGY

Fine-Grained Vehicle Detection (FGVD) aims to localize the vehicles in an on-road scene image and identify their type, manufacturer, and model variant. We accomplish this in two stages: the first stage involves vehicle localization, and the second involves fine-grained classification of the localized object. The entire pipeline is shown in Fig. 6. In the localization stage, we use YOLOv5 [5] model, which gives us the vehicle bounding boxes. We then crop out the vehicles’ Regions Of Interest (ROIs) from the original image using the bounding boxes obtained in the localization stage. The cropped ROIs are then resized before feeding them to the classification module. In the classification module, we use the Label Relation

Graphs Enhanced Hierarchical Residual Network (HRN) model [2], which predicts the coarse to fine-grained classes for the ROIs.

### 4.1 Vehicle Localization

We train the YOLOv5 model to localize the vehicles in the FGVD dataset<sup>3</sup>. There are various reasons for choosing the YOLOv5 model for vehicle localization. Firstly, YOLOv5 incorporates Cross Stage Partial Network (CSPNet) [20] into its backbone and in the neck. The CSPNet helps to achieve a richer gradient combination while reducing the amount of computation, which ensures the inference speed and accuracy are high and reduce the model size.

Moreover, the HRN’s classification accuracy depends on the localization model’s performance; thus, maintaining high accuracy for vehicle localization is essential. Secondly, the head of YOLOv5 generates three different sizes ( $18 \times 18$ ,  $36 \times 36$ ,  $72 \times 72$ ) of feature maps to achieve multi-scale [13] prediction, enabling the model to handle small, medium, and large-sized objects. YOLOv5 also auto-learns custom anchor boxes such that the anchors are adapted to our FGVD dataset, which helps improve the detection results. It also incorporates various augmentations, such as mosaic, during training which significantly helps to generalize. Moreover, we experiment with Faster-RCNN [14], but we obtain the best results with YOLOv5 (refer to Sec. 6) while it also takes the least training time.

### 4.2 Fine-grained Classification

For several reasons, we chose the Label Relation Graphs Enhanced HRN network [2] for fine-grained vehicle classification. The HRN

<sup>3</sup>For training the HRN model, we prepare the dataset separately by cropping out vehicle images using the ground truth boxes from our dataset.

network focuses on encoding the label hierarchy from coarse-to-fine levels. The HRN accomplishes this by using the Granularity Specific Blocks (GSB) and residual connections, as shown in Fig. 4. Each GSB block extracts the hierarchical level features by processing the feature maps generated from the trunk network, i.e., any common feature extraction network pre-trained on ImageNet [3]. The residual connections combine the features of coarse-level and fine-level subclasses. This kind of hierarchical modeling primarily benefits the FGVD application on our dataset because there are many similarities between different vehicle model variants corresponding to the same manufacturer or vehicle type. The HRN model incorporates a combinatorial loss which aggregates information from related labels defined in the tree hierarchy. This tree hierarchy uses a sigmoid node for each label, which can be seen in Fig. 6 for the L-1, L-2, and L-3 outputs. The HRN models independent relations using sigmoid instead of softmax since sigmoid implies mutual exclusion. But if the training samples at the fine-grained levels are few, the combinatorial loss would fail to well-separate the skewed classes. So, an additional multi-class cross-entropy loss is used with the softmax function for the finest labels, depicted for L-3 class output in Fig. 6. The softmax function increases the weightage of fine-grained classification loss, ultimately ensuring high classification accuracy specifically for the fine-grained labels. Moreover, the residual connections in the HRN for hierarchical feature interactions make the architecture effective compared to other models while demonstrating state-of-the-art performance on standard fine-grained classification datasets.

We also create a hierarchical tree structure of labels for each of our fine-grained classes in the format required by the HRN architecture. It is important to note that we use the softmax output for the fine-grained class instead of the sigmoid output in the HRN model at inference time. As mentioned by Chen et al. [2], the softmax output channel computes separate cross-entropy loss so that the mutually exclusive fine-grained classes gain more attention during training.

## 5 EXPERIMENTS

We split our entire dataset into train:val:test ratio of 64:16:20. We use the YOLOv5l [5] model pre-trained on the COCO dataset [8] and fine-tune it on the FGVD dataset for 100 epochs with a batch size of 8. The input images are pre-processed and resized to  $640 \times 640$  pixel dimensions before feeding them to the training pipeline. While training, we observe that the objectness loss on the validation set reaches the lowest point after a few epochs and then starts diverging. To resolve this, we reduce the contribution of the objectness loss to the overall loss function by half. We train the HRN model on the ground truth Regions of Interest (ROIs) and all three FGVD levels. We use the resnet-50 [4] model, pre-trained on ImageNet [3], for the trunk net in the HRN architecture. The input image size used here is  $448 \times 448$  pixel dimensions. We train the HRN model for 100 epochs with a batch size of 8 and an initial learning rate of 0.001. At test time, we use the trained YOLOv5l model for predicting the vehicle bounding boxes and then use the predictions to crop ROIs for HRN's input.

We also experiment with two baseline detectors for the FGVD task. Firstly, we train the Faster-RCNN model on the FGVD's level-3

labels for 100 epochs with a batch size of 8 and an initial learning rate of 0.001. We use the Faster-RCNN with the resnet-50 backbone, and the model is pre-trained on the COCO detection dataset while taking an input image size of  $512 \times 512$ . Secondly, compared to the Faster-RCNN detector model, we train a similar sized YOLOv5-large variant model on level 3 labels for 100 epochs with a batch size of 16. We train the YOLOv5l model with the same hyperparameter configuration and the same pre-trained model as used in the first experiment explained above.

We evaluate the performance of our models using the mean Average Precision (mAP) metric on all three levels. For the baselines, we derive the mAPs for all the levels from the combined label (sample combined labels are shown below the zoomed-out vehicle crops in Fig. 1 a) on which the detectors are trained. We use GeForce GTX 3080 Ti GPU for all our experiments. We present the results of the above experiments in the next section.

**Table 3: FGVD detection results (in %): The combination of YOLOv5l and HRN significantly improves the mAPs on all levels compared to existing detectors.**

Model	L-1 mAP	L-2 mAP	L-3 mAP
F-RCNN	54.43	41.46	31.92
YOLOv5l	61.70	42.40	32.75
<b>YOLOv5L + HRN</b>	<b>83.21</b>	<b>59.02</b>	<b>48.40</b>

## 6 RESULTS

We show the results of our detection experiments in Table 3. As shown in its first row, with the Faster-RCNN model, we obtain the mAPs of 58.9%, 41.4%, and 31.9% on level 1 (L-1), level 2 (L-2), and level 3 (L-3) labels, respectively. The YOLOv5l model obtains mAPs of 61.7%, 42.4%, and 32.7% on the three levels. We observe that the YOLOv5l detector performs significantly better than the Faster-RCNN model on all the hierarchical levels. We thus choose the YOLOv5l model with HRN for the final experiment. Using the YOLOv5l+HRN model, the mAP scores at L-1 and L-2 levels also improve by 21.5% and 16.6%, respectively. We obtain an mAP of 48.4% on the L-3 level, which is a substantial performance boost of 15.7% compared to the standard YOLOv5l model.

We showcase the YOLOv5l+HRN model's detection results on sample images from the FGVD test set in Fig. 7. The classified fine-grained labels from the YOLOv5l+HRN model are shown alongside the predicted bounding boxes. The correctly classified predictions are shown in green, and the incorrect predictions are in red. The figure depicts our model's performance on images containing vehicles at multiple viewpoints and resolutions with occlusions and low visibility scenarios. It can be observed that the model is giving correct predictions for the occluded vehicles with low visibility, like the Hyundai Santro car in the left image of Fig. 7. Similarly, the Honda Shine motorcycle on the right image left side of the same figure is classified correctly at all three levels. We also showcase an occlusion scenario where the model gives an incorrect prediction. The right image in Fig. 7, shows that for the Hyundai Creta car (ground truth), our model gives incorrect prediction as Hyundai Santro car, which is highlighted in the red box. The zoomed-out views with adjusted brightness show the actual vehicle images

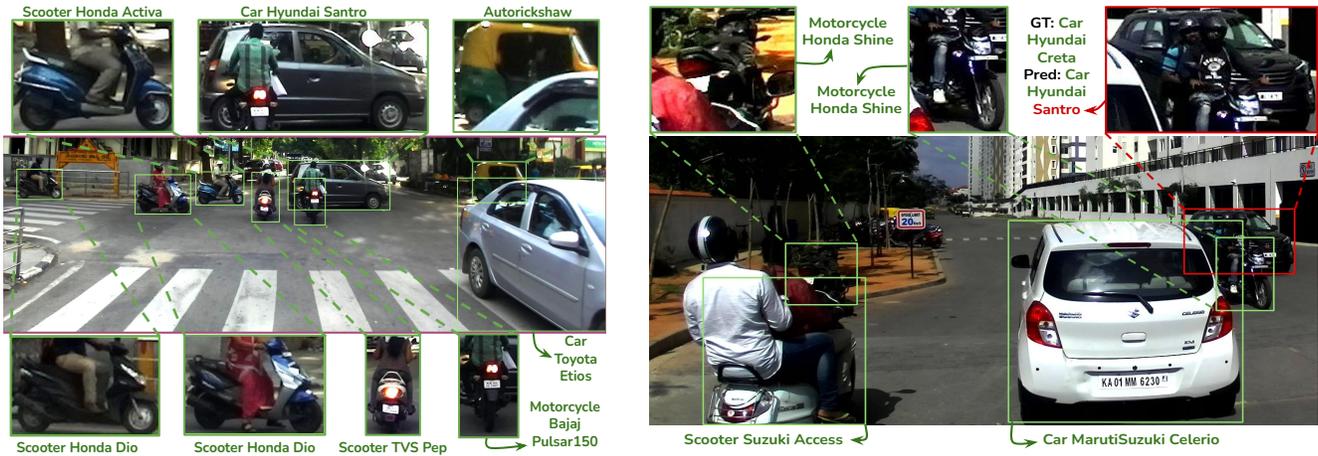


Figure 7: FGVD results of the YOLOv5l+HRN model on sample scene (crops shown for better visibility) images from the test dataset. The detected bounding boxes are marked in green and red, corresponding to correct and incorrect class predictions. The zoomed-in views highlight the vehicles which are occluded or have low visibility.

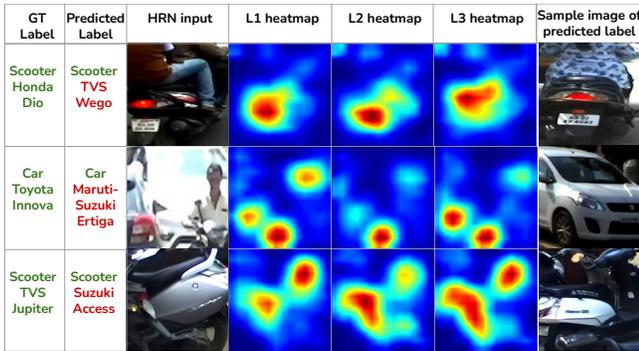
L1 (Vehicle Type)	Scooter				Motorcycle				Car			
L2 (Manufacturer)	TVS	Suzuki	TVS	Suzuki	RoyalEnfield	TVS	RoyalEnfield	Bajaj	TataMotors	Toyota	Toyota	MarutiSuzuki
L3 (Model)	Jupiter	Access	Jupiter	Access	Classic350	ApacheRTR 160	Classic350	Avenger	Indica	Innova	Innova	Dzire
Yolov5l detected vehicle as HRN input												
L1 GradCAM++												
L2 GradCAM++												
L3 GradCAM++												

Figure 8: Rows 1-3: correct predictions/labels obtained from YOLO+HRN with ROIs in row 4. GradCAM++ [1] visualizations on L1, L2, and L3 levels (rows 5-8) from the HRN network, when applied on the row 4 ROIs. Multiple orientations of scooters, motorcycles, and cars are compared. From Row 4, Col 2: two front-view images of scooter models, followed by the rear-view images of the same models. Row 4, Col 6: two side-view motorcycle model images followed by two rear-view images. Row 4, Col 10: two side-view and two back-view images of different car models. The GradCAM++ visualizations highlight the distinguishing features for each ROI, like the front, side, and rear body structures, backlight/blinker (most vehicles), silencer (Access rear-view and Classic 350), petrol tank (Jupiter rear-view), door handles, and engine shape for the corresponding vehicles.

clearly. For the incorrect detection in Fig. 7, we can observe that the model detects vehicle type and manufacturer correctly, even in the high occlusion scenario.

In Fig. 8, we showcase the GradCAM++ [1] visualizations of the detected fine-grained vehicles from the YOLO-HRN model. On the figure’s left side, we compare two front-view scooter images. The L1 GradCAM++ heatmap of the Jupiter model shows the model’s focus on most of the front body parts along with its blinkers. The corresponding heatmap for the Access model also shows the focus on the front body part along with the mudguard region. While the L3 visualizations focus on very particular attributes for the Jupiter

model, it is centered around the air vent part above the mudguard. For the Access model, the focus is on the shape of the front body part. Similarly, we compare the rear views of these vehicle models. The L1 heatmap of the Jupiter model shows a major focus on the rear body part with some highlights (in green) on its speedometer display. In contrast, for the Access sample, the primary focus is on the back body part with some highlights around the silencer, speedometer display, and the key region. Overall L1 heatmaps show the model’s focus on most of the crucial body parts of the vehicle. While in the L3 visualizations of the Jupiter model the focus is centered around the petrol tank lid area just above the backlight,



**Figure 9: Failure cases with their GradCAM++ [1] visualizations for L1, L2, and L3 levels, along with images of an incorrectly predicted vehicle model for comparison (on the right) with the actual input image (on the left). Top and Bottom: Examples of high similarity in visual cues between the predicted and the actual classes. Mid: extreme illumination with occlusion from multiple objects.**

and for the Access, it is primarily centered around the backlight region with some highlights on the silencer part as well. These features highlighted by the L3 visualizations are the main feature points that help distinguish the two vehicle models.

We have made similar comparisons between motorcycles and cars. We first compare the side views of the two models and then the rear views of the other two models. The L1 heatmap of the Classic350 model’s side view shows the model’s focus primarily around the suspension area just beside the triangular engine boundary, with some focus on the petrol tank, silencer, and the engine area at the bottom. While the L1 visualizations of the ApacheRTR160 TVS motorcycle focus on the vehicle’s back body part, the back wheel’s central area, and the front wheel. The L1 visualizations cover all the important key points that help identify a motorcycle, along with providing additional feature points for the classification at finer levels. The L2 visualizations highlight regions very similar to L3 visualizations, which ultimately help identify the correct L3 label. The L3 visualizations of the Classic350 model show the main focus on the suspension region and some on the silencer. For ApacheRTR160, the main focus is around the back body part. These focus points correspond to the unique feature points which are specific to that vehicle model only. Now we compare the rear views of Classic350 and Avenger models. These models’ L1 and L3 heatmaps do not show much difference since only the back region of these vehicles are visible, where the number of critical key points is limited. The main distinguishing key point which is being highlighted in these heatmaps is around the backlight region and a little bit on the side body part, which is partially visible.

Similarly, we compare the side views of cars. The L1 and L3 visualizations of the side views of the Indica and Innova models look nearly similar. For Indica, the door handle and the front window pillar regions are primarily focused, while the entire side body part is also highlighted in green. For Innova, the door handles and the window edges are primarily focused. Similarly, the rear views of Innova and Dzire show a primary focus on the shape and structure of

**Table 4: Recognition accuracies (in %) of HRN on standard fine-grained classification datasets and FGVD. For all three levels, the FGVD dataset remains the most challenging.**

Dataset	L-1	L-2	L-3	# cls.	# objs.
Aircraft [9]	97.45	95.79	92.58	100	10000
CUB-200-2011 [19]	98.67	95.51	86.60	200	11788
Stanford Cars [7]	97.41	94.03	NA	207	16185
<b>FGVD dataset</b>	<b>96.69</b>	<b>79.44</b>	<b>76.35</b>	<b>217</b>	<b>24450</b>

the back body part along with the emphasis on the backlight region as visible in the Dzire model. Again due to the limited number of key points, we suspect that the highlighted parts in all the levels are similar, highlighting the entire overlapping regions considered for the L3 classification. Overall we showcase the hierarchical relationship between the coarse-level and fine-level labels and the overlap of similar features among them. Fig. 9 shows the failure cases for similar vehicles in extreme illumination/occlusion scenarios.

We also evaluate the L1, L2, and L3 label’s classification accuracies using the fine-tuned HRN model on our test dataset and compare it with the corresponding state-of-the-art accuracies on standard fine-grained classification datasets as reported in the paper [2]. We find that the Level-3 classification accuracy on our dataset is the lowest, i.e., 76.35%, among other datasets, despite having more objects in FGVD, as shown in Table 4. The reason for low accuracy on more fine-grained levels of FGVD is the high diversity in classes (217) and complexities like occlusions, non-object-centric images, pose variations, scale variations, lighting conditions, etc. The results quantitatively demonstrate the high complexity involved in the FGVD dataset and the challenges involved in the classification and detection of fine-grained vehicles in the wild.

Although the YOLOv5l+HRN model performs better than other baseline detectors, there is still a massive gap of around 35% mAP values between the coarse-level and the fine-grained level detection performances in Table 3. Thus, we propose this dataset for future research works related to the FGVD.

## 7 CONCLUSION

This paper presents the first dataset for the Fine-Grained Detection of Vehicles while also providing the baselines for the same. FGVD is also the first dataset in which the fine-grained labels for 5 additional vehicle types are available apart from just cars. Our dataset can be used for vehicle re-identification in on-road surveillance systems, generating alarms for road safety systems, and promoting the development of ADAS products for Indian roads. Specifically, we showcase the uniqueness of our dataset regarding the detection of fine-grained vehicles in the wild compared to other related datasets. We also provide the results of the YOLOv5l+HRN model for the FGVD dataset, with which we obtain a 15.7% gain in mAP over baseline detectors for the most complex level. For future works, we plan to fuse the architectures of detection and classification models to improve the overall efficiency of the solution.

**Acknowledgement:** We thank iHubData, the Technology Innovation Hub (TIH) at IIT-Hyderabad for supporting this project.

## REFERENCES

- [1] Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, and Vineeth N. Balasubramanian. 2017. Grad-CAM++: Generalized Gradient-based Visual Explanations for Deep Convolutional Networks. *CoRR* abs/1710.11063 (2017). arXiv:1710.11063 <http://arxiv.org/abs/1710.11063>
- [2] Jingzhou Chen, Peng Wang, Jian Liu, and Yuntao Qian. 2022. Label Relation Graphs Enhanced Hierarchical Residual Network for Hierarchical Multi-Granularity Classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4858–4867.
- [3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 248–255. <https://doi.org/10.1109/CVPR.2009.5206848>
- [4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. *CoRR* abs/1512.03385 (2015). arXiv:1512.03385 <http://arxiv.org/abs/1512.03385>
- [5] Glenn Jocher, Alex Stoken, Jirka Borovec, NanoCode012, ChristopherSTAN, Liu Changyu, Laughing, tkianai, Adam Hogan, lorenzomamma, yxNONG, AlexWang1900, Laurentiu Diaconu, Marc, wanghaoyang0106, ml5ah, Doug, Francisco Ingham, Frederik, Guilhen, Hatovix, Jake Poznanski, Jiacong Fang, Lijun Yu, changyu98, Mingyu Wang, Naman Gupta, Osama Akhtar, PetrDvoracek, and Prashant Rai. 2020. *ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements*. <https://doi.org/10.5281/zenodo.4154370>
- [6] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. 2011. Novel Dataset for Fine-Grained Image Categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*. Colorado Springs, CO.
- [7] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3D Object Representations for Fine-Grained Categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*. Sydney, Australia.
- [8] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. *CoRR* abs/1405.0312 (2014). arXiv:1405.0312 <http://arxiv.org/abs/1405.0312>
- [9] Subhansu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151* (2013).
- [10] Syeda Aneeba Najeed, Rana Hammad Raza, Adeel Yusuf, and Zamra Sultan. 2022. Fine-grained vehicle classification in urban traffic scenes using deep learning. In *Proceedings of the 11th International Conference on Robotics, Vision, Signal Processing and Power Applications*. Springer, 902–908.
- [11] Maria-Elena Nilsback and Andrew Zisserman. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*. IEEE, 722–729.
- [12] Joseph Redmon and Ali Farhadi. 2018. YOLOv3: An Incremental Improvement. (04 2018).
- [13] Joseph Redmon and Ali Farhadi. 2018. YOLOv3: An Incremental Improvement. *CoRR* abs/1804.02767 (2018). arXiv:1804.02767 <http://arxiv.org/abs/1804.02767>
- [14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1 (Montreal, Canada) (NIPS'15)*. MIT Press, Cambridge, MA, USA, 91–99.
- [15] J. Sochor, J. Špaňhel, and A. Herout. 2018. BoxCars: Improving Fine-Grained Recognition of Vehicles Using 3-D Bounding Boxes in Traffic Surveillance. *IEEE Transactions on Intelligent Transportation Systems* PP, 99 (2018), 1–12. <https://doi.org/10.1109/TITS.2018.2799228>
- [16] Pei Sun, Henrik Kretschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. 2020. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [17] Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. 2015. Building a Bird Recognition App and Large Scale Dataset With Citizen Scientists: The Fine Print in Fine-Grained Dataset Collection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [18] Girish Varma, Anbumani Subramanian, Anoop Namboodiri, Manmohan Chandraker, and CV Jawahar. 2019. IDD: A Dataset for Exploring Problems of Autonomous Navigation in Unconstrained Environments. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 1743–1751.
- [19] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. 2011. The caltech-ucsd birds-200-2011 dataset. (2011).
- [20] Chien-Yao Wang, Hong-Yuan Mark Liao, I-Hau Yeh, Yueh-Hua Wu, Ping-Yang Chen, and Jun-Wei Hsieh. 2019. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. *CoRR* abs/1911.11929 (2019). arXiv:1911.11929 <http://arxiv.org/abs/1911.11929>
- [21] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. 2015. A Large-Scale Car Dataset for Fine-Grained Categorization and Verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [22] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. 2020. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2636–2645.