Learning to Detect Objects at Multiple Orientations and Occlusion Levels

Eshed Ohn-Bar and Mohan Manubhai Trivedi, *Fellow, IEEE*

***Abstract*—We study efficient means of capturing intra-category diversity for object detection. Strategies for occlusion and orientation handling are explored by learning an ensemble of models using visual and geometrical features. An AdaBoost detection scheme is employed with pixel lookup features for achieving fast detection. The method show promise in terms of detection performance and orientation estimation accuracy on the challenging KITTI dataset.**

***Index Terms*—Object detection, multiorientation detection, mining appearance patterns, occlusion-handling, vehicle detection, pedestrian detection, active safety, orientation estimation, performance evaluation.**

I. Introduction

Efficient object detection requires robustness to the appearance variations of an object. In the context of vehicle detection studied in this work, these variations may stem from a changing observation angle, illumination variability, vehicle shape and type, truncation out of the camera view, different occlusion levels, etc.

Many recent successful frameworks in dealing with such challenges build upon the deformable parts model (DPM) [1]. The DPM employs three main elements; deformable parts in a pictorial structure, latent discriminative learning, and a multi-components mixture model. A main question we deal with in this work is what is the best approach for dividing the training data in order to learn the multi-component model. Although other aspects of the DPM may also provide detection robustness, the emphasis for studying components is well motivated. Multiple components provide a natural accommodation of object appearance variation due to geometry, orientation, and occlusion. In specific object detection domains, such as vehicle or pedestrian detection, certain appearance patterns (such as certain occlusion types) may be common, thereby motivating learning components for such patterns. Furthermore, several recent studies show components to be useful in varying domains of object detection [2]–[5].

The preferable approach for obtaining the subcategory component clusters is not trivial. For instance, this may be done by aspect ratio of bounding boxes or visual cues [3]. Common solutions employ a latent discriminantive clustering process (latent SVM) which may produce degenerate or noisy category clusters [6]. A main limitation of the classical DPM framework lies in several speed bottlenecks. This further motivates our study, as subcategory models are expensive to evaluate in test time. Therefore, learning better subcategory models could result in speed gains without hindering detection performance. This aspect of the detection scheme is of particular interest in mobile settings of intelligent transportation systems, where fast and lightweight computation is desired.

In this work, we study object subcategorization using clustering of 3D orientation, position, occlusion level and types, and other geometrical shape features. This study demonstrates the following:

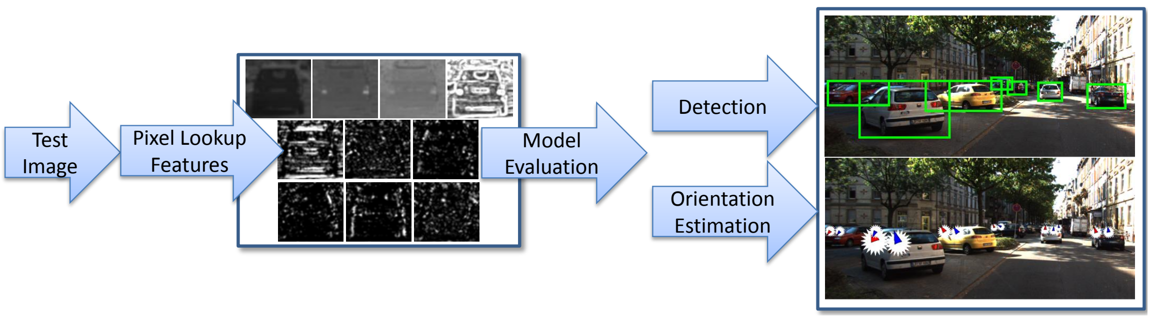
Features for clustering of object subcategories: Learning good subcategory models is shown to be highly dependent on the features used. In particular, it is shown to work best (i.e. produce homogeneous clusters useful for detection) when using a set of 3D features and annotated features. Furthermore, clustering techniques are studied and compared in terms of final detection quality. The following questions motivate the study of this paper: How should one quantize the data best in order to obtain good subcategory models? How to efficiently employ such a framework to handle occlusion and orientation variation? Should model learning occur over varying occlusion levels, if so, how should these be chosen? Should we consider statistics of the occluder and different occlusion types? What if no 3D orientation information is available? How does the choice of subcategories affect orientation estimation? A main contribution of this work is in providing analysis for such questions.

Speed and merging of detection approaches: Most recent work in vehicle detection has been focused on DPM and HOG-based detectors, which contain several speed bottlenecks in its classical implementation and formulation. At the same time, pedestrian detection has seen considerable speedups [7], [8]. The work in this paper can be seen as an attempt to merge the two; it aims to achieve fast object detection using integral features with a soft-cascade framework but without compromise in detection accuracy when compared to the DPM. Depending on the number of subcategories, the entire detection pipeline can run between 13 (for 1 model) to 5 (for 20 models) frames per second (fps) on full resolution images of size 1242×375, and further speedups are possible [9], [10]. The scheme outperforms the DPM baseline in performance, and is significantly faster (the DPM runs at about 10 seconds per frame).

Orientation estimation: After optimization of the features, clustering techniques, and number of subclusters, these are used to perform highly robust orientation estimation. This task summarizes the effectiveness of the framework. The framework is also shown to generalize to other detection tasks, such as pedestrian detection. Fast detection and recognition of objects in the scenes is essential in development of intelligent vehicles applications, for instance in trajectory understanding [11].

The authors are with the Laboratory for Intelligent and Safe Automobiles (LISA), University of California San Diego, San Diego, CA 92093-0434 USA (e-mail: eohnbar@ucsd.edu; mtrivedi@ucsd.edu).

(a) Training pipeline.



(b) Testing pipeline.

Fig. 1: Key components of the learning and detection framework studied in this paper. Subclusters with visual homogeneity are extracted using a set of visual, occlusion type, orientation, or other geometrical features. In (a), the results of the clustering using 3D geometry features is shown. The clusters provide positive training samples for a cluster-specific model. For instance, the average image of two example clusters are shown at a certain orientation. In testing, a set of pixel lookup features is derived from color and gradient cues providing fast feature extraction. Schemes are evaluated in terms of detection performance and orientation estimation accuracy.

II. Related Research Studies

Commonly, sliding window-based vehicle detection may employ a variant of HOG+SVM (histogram of oriented gradients and a linear support vector machine) [18] or cascade detectors [7]. We briefly review recent relevant literature, and turn the reader to the comprehensive review of [19] for additional detail.

Vehicle detection with DPM: The DPM model [1], which builds on HOG features and a Latent SVM, has also been a common choice for vehicle detection [13], [20]. In [14], a variant of the DPM framework is used in order to detect vehicles under heavy occlusion and clutter. In [20], integrating scene information was shown to improve both the detection and orientation estimation performance of the DPM. To better handle detection of occluded vehicles, a second-layer conditional random field (CRF) was used over root and part score configurations provided by a DPM model in [21]. AND-OR structures were employed in [22] to detect highly occluded vehicles from parts. In the aim of detecting objects under occlusion, a joint object detector was proposed in [23], and bounding boxes were predicted using linear regression. Although the approach in [23] appears promising for pedestrian detection, the study of [4] showed a joint vehicle detector with bounding box regression to perform worse than a single object DPM detector. Nonetheless, a main improvement over the DPM baseline was gained by incorporating mixture components specifically for occluded vehicle cases. This motivates our study of learning models for appearance variations due to occlusion and other geometrical and visual features. In [4], [23], [24], an arbitrary partition of the data is performed to produce an initialization to the LSVM-based assignment. These studies generally consider a subset of the geometrical features studied in this work. Furthermore, the choice of subcategorization features and techniques were not present. Interestingly, in our study initialization of a discriminative clustering of visual data framework (e.g. LSVM) with different geometrical features resulted in only minor improvements. As a matter of fact, only working in the 3D geometry space produced the best results in terms of detection performance. To emphasize, unlike the aforementioned studies, we also study the impact of different features and clustering techniques on the final detection. We show that with the right features, both k-means or spectral clustering [25] produce a good quantization of the visual feature space for training detection models.

Subcategory learning: A common approach for improving model generalization is by learning subcategories within an

TABLE I: Overview of related research studies for vehicle detection at multiple orientations and occlusion levels.

Study	Features	Classifier	Subcategory Clustering	Subcategory features	Parts	Occlusion-Handling	Speed (fps)
Kuo and Nevatia [12] (2009)	HOG	GentleBoost	LLE	HOG	Y	N	-
Niknejad *et al.* [13] (2012)	HOG	LSVM	LSVM	Aspect-ratio	Y	N	$\sim$0.5 (640 $\times$ 480)
Hejrati and Ramanan [14] (2012)	HOG	LSVM	k-means/EM	Part configuration and occlusion type	Y	Y	0.03 (1242 $\times$ 375) †
Pepik *et al.* [4] (2013)	HOG	LSVM	Rule-based	3D orientation and occlusion types	Y	Y	0.1 (1242 $\times$ 375)
Li *et al.* [15], [16] (2013)	HOG	AND-OR structure	AND-OR Tree	Aspect-ratio and occlusion	Y	Y	-
Sivaraman and Trivedi [17] (2013)	Haar	AdaBoost	-	-	Y	Y	14.5 (500 $\times$ 312)
This study	Color, gradient orientation, and gradient magnitude	AdaBoost	Unsupervised and Supervised	Geometrical and visual features	N	Y	5 fps (1242 $\times$ 375) ‡

†: Not reported in the paper but produced by us using the publicly available code.
‡: Run-time depends on the number of subcategories. This number is for 20 models.
Index: **LSVM**: Latent Support Vector Machine. **EM**: Expectation Maximization. **HOG**: Histogram of Oriented Gradients.
fps: Frames per Second. **LLE**: Locally Linear Embedding.

object class. For instance, these are used in conjunction with DPMs in order to detect objects at varying aspect ratios. In [12], visual subcategories corresponding to vehicle orientation are learned in an unsupervised manner using Locally Linear Embedding and HOG features. In [26], an exemplar SVM is learned for each positive example, and the learned weights are used in affinity propagation to generate visual subcategories. This exemplar-based step provides the initialization to LSVM clustering. Adding mixture components for occluded objects was shown to be promising in [4]. Several other recent studies have shown the benefit of visual homogeneity in training data on model performance [27]. Vehicle orientation estimation is studied using supervised, semi-supervised, and unsupervised settings with DPM framework in [28], with supervised settings showing the best results.

Discriminative subcategorization, where the clustering considers negative instances, was studied in [6]. The technique is shown to improve results over Latent SVM-based clustering refinement, which is shown to be prone to cluster degeneration. Furthermore, the framework in [6] provided more visually consistent clusters. This technique will be used as a baseline in our work.

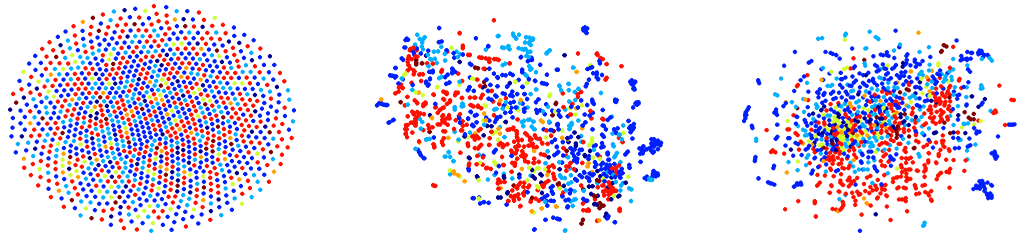
In [3], the importance of efficient learning of visual subcategories for different objects is highlighted. By using an extension of the Latent SVM framework initialized with k-means on visual data, a significant gain in performance was shown on the PASCAL dataset. The authors in [3] motivate visual subcategorization over other forms of data partitioning by arguing that tighter clusters can be extracted from visual data, as semantic (human-based) subcategories simply aim to encode visual consistency.

One important advantage of such an approach is that no annotation is required. While this may be correct, we argue that this may substitute a hard problem (of annotation) with another hard problem of visual subcategorization. Furthermore, despite the motivation for subcategorization using visual features, no direct comparison of the impact that different features have on clustering was performed in the aforementioned works. In this paper, we use the KITTI dataset [29] to answer such questions, and visual subcategorization is shown to be significantly inferior to geometrical 3D orientation and occlusion features.

Table I outlines the differences between existing approaches and ours. We pursue an alternative approach to the DPM. In [17], front and rear parts are detected, tracked, and associated to produce a vehicle bounding box. By using Haar-like features with AdaBoost, fast detection speed is achieved. We also pursue fast detection, by using a wider range of features which can also be computed in real-time [7]. Furthermore, we encode occlusion without a notion of parts by learning a specific model for occluded vehicles. Different approaches are also compared in Table I, yet we note that reported running times vary based on image size and other parameters. For instance, in [13] a limit is set on the minimum size of the detcted objects which could allow for a speedup. For the framework studied in this work, the main parameter is the number of subcategories. Since each subcategory requires a separate model evaluation, the trade-off between speed and performance will be analyzed in Section VII.

III. Object Subcategorization

The key components of the proposed framework are shown in Fig. 1. Ultimately, the goal is to learn visually homogeneous clusters, which, due to less intra-cluster ambiguity, produce better models as opposed to, say, learning one model over all instances (a monolithic classifier). In the case that 3D orientation and geometry features are not available, a vision scientist may work with visual data only. A clustering algorithm is used to produce a predetermined number of clusters, K. We first detail the visual subcategorization studied in this work,



(a) RGB-Color (b) LUV-Color (c) Gradient

Fig. 2: Visualization of the feature space using t-SNE [30] with varying features on the entire KITTI training dataset (some samples were suppressed for visualization). The color of each point corresponds to an assigned bin according to annotated vehicle orientation. We note that both color and gradient (orientation and normalized magnitude) cues provide useful information for detection and subcategory clustering.

because most of the later analysis involves the fine-grained 3D geometry features that are available for the KITTI dataset.

A. *Visual Clustering*

As shown in Table I, shape features are commonly used for capturing the shape patterns of vehicle instances. We studied both gradient (orientation and normalized magnitude) and color features, as shown in Fig. 2. In Fig. 2, each point is colored according to its 3D orientation. It is observed that both color and gradient features to correlate with vehicle orientation. Since orientation accounts for much of the appearance variation, both cues are used for subcategorization. Pixel values in LUV color space are shown to be more useful when compared to the RGB color space. Color may capture cues such as taillights. A total of 10 types of features is used for visual subcategorization: LUV color, normalized gradient magnitude, and oriented gradients at 6 bins (as in [7]). These 10 feature types were all shown useful for detection of vehicles in our experiments, and can be extracted at more than 55 fps on a CPU (6 core, Intel Core i7 @ 3.30 GHz with 16 GB RAM) for full resolution images of size 1242×375. As pre-processing, all positive instances are resized to the mean image size $[w_m, h_m]$. Next, the 10 feature types are extracted and downsampled by a factor of 4 for a descriptor of size $w_m \cdot h_m \cdot 10/16$.

B. *Mining Object Geometry, Orientation, and Occlusion Patterns*

Generally, as visual clustering is challenging, geometry features are integrated into the clustering. We first detail the types of features that will be studied in this work, and consequently the different strategies that will be employed to produce clusters using the features. 3D Object information can be estimated from the scene using either manual annotations, stereo, or another sensor, such as a Velodyne lidar which was used to generate 3D ground truth information for the KITTI dataset. In the latter case, an object in the lidar coordinates can be annotated and projected to a point in the camera image using a rigid body transformation. Let $\mathbf{R}_\mathbf{l}^\mathbf{c} \in \mathbb{R}^{3\times3}$ and $\mathbf{t}_\mathbf{l}^\mathbf{c} \in \mathbb{R}^{1\times3}$ be the lidar to camera rotation matrix and translation vector respectively, obtained using [31]. Then a 3D point, $\mathbf{x} = (x, y, z, 1)^T$ in the lidar coordinates is projected to the image point $\mathbf{y} = (u, v, 1)^T$ using

$$\mathbf{y} = \mathbf{PRTx} \tag{1}$$

with the projection matrix of the camera $\mathbf{P}$, a 4×4 rectifying rotation matrix $\mathbf{R}$, and

$$\mathbf{T} = \begin{pmatrix} \mathbf{R}_\mathbf{l}^\mathbf{c} & \mathbf{t}_\mathbf{l}^\mathbf{c} \\ 0 & 1 \end{pmatrix} \tag{2}$$

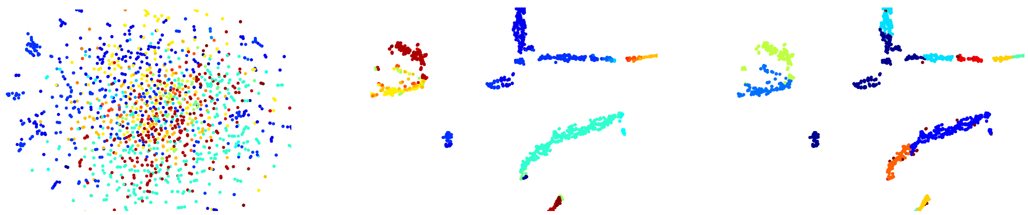
The availability of high quality 3D information raises the following research question: can these be used to learn detection models, as opposed to visual features? How should these different modalities be integrated to produce the best models? These questions are the main study of this work.

To represent vehicle instances, we extract the following set of geometrical features.

3D orientation: When detecting vehicles in different driving environment (intersection, highway, etc.), appearance variation due to the observation angle is therefore common. Instead of using the raw 3D yaw angle (rotation around the Y-axis in camera coordinates), the observation angle is used (the angle of the vector joining the camera center in 3D and the other object). The reason for this is that the yaw angle as it is does not take into account the ego-vehicle, which may be observing the object from different angles. For instance, an object at 90 degrees may appear very differently depending on where it is located around the ego-vehicle.

Aspect-ratio: The 2D bounding box of objects is correlated with the geometry of the object being detected. We explicitly include this in the clustering, with the aim of creating a different model for objects at different aspect ratios. These may not necessarily involve different orientations (e.g. a car vs. a bus). Learning models at different aspect ratios provides a significant improvement in detection (as opposed to keep a fixed dimension model for all clusters).

To encode variation in appearance due to occlusion, it may not be necessary to train a model for very fine-grained occlusion levels. As a matter of fact, training a model on heavily occluded objects was not shown to be successful in our experiments. Nonetheless, as occlusion level is a main factor in visual diversity, we found that using explicit occlusion features



(a) Visual Features (b) Geometrical Features (c) Clustering Output

Fig. 3: (a) and (b): The color of each point corresponds to a quantization into 10 bins in orientation space and 2 bins in occlusion space (occluded and non-occluded). In (a), color and gradient features (see Section III-A) are used for the t-SNE visualization. In (b), the proposed set of geometrical features (see Section III-B). (c) Output of k-means clustering, which can be used for training the subcategory models.

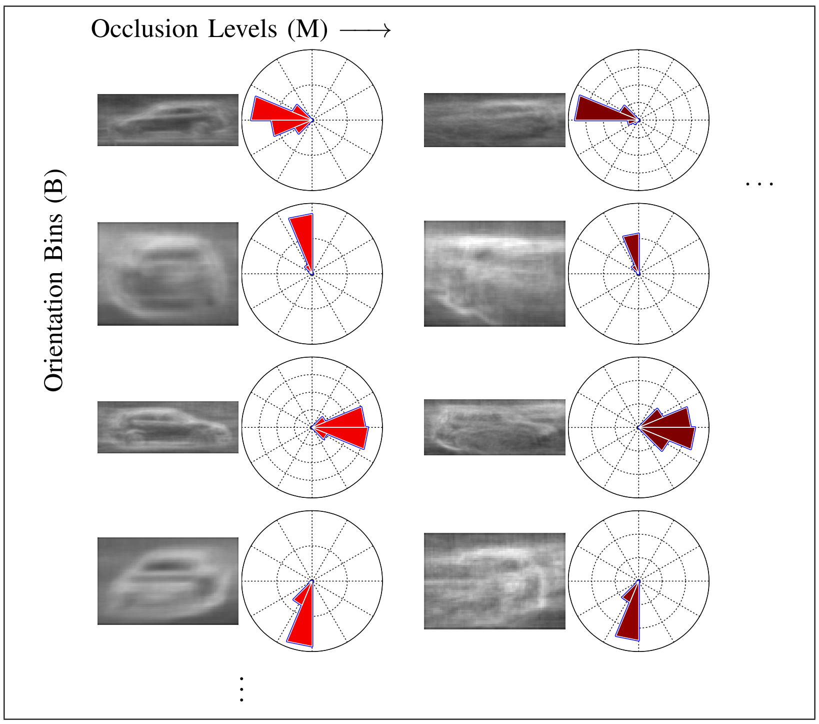


Fig. 4: We study strategies for object detection at multiple orientation and occlusion levels. One possibility is to cluster the data and learn specific orientation and occlusion models suitable for the appearance variations within that cluster. We visualize the mean gradient image in each cluster. Darker rose plots correspond to higher number of occluded samples in the cluster.

in the clustering can improve detection of partially-occluded vehicles.

Truncation level: The percentage of the vehicle outside of the camera view is also used as a feature.

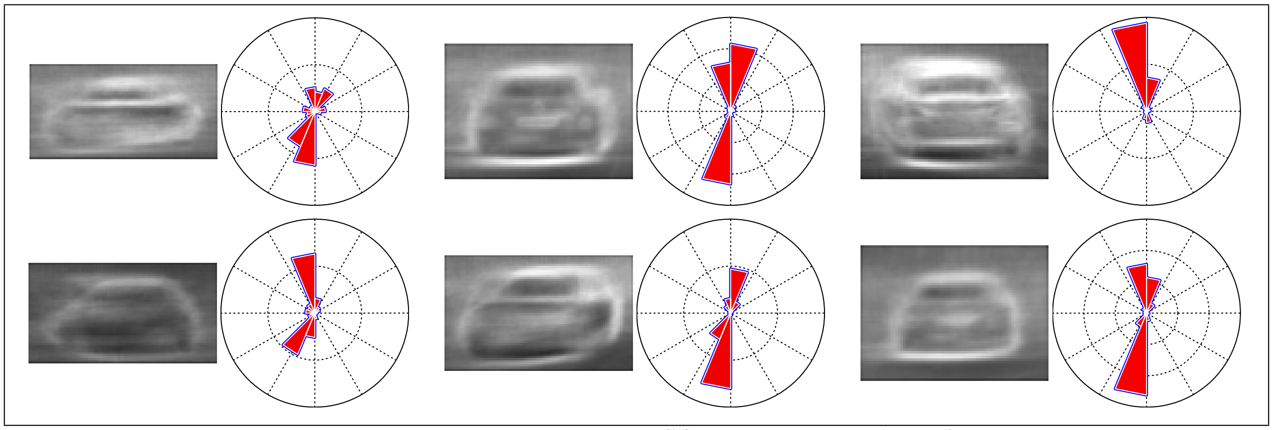
Occlusion level: Given a 2D bounding box, we search the 3D space for an occluder. The closest vehicle in 3D that is also closer to the camera than the occluded vehicle. The occlusion degree feature is the overlap of the two 2D bounding boxes, overlap($BB_{occluder}$,$BB_{occludee}$), where overlap($b1$,$b2$) $= \frac{area(b1 \cap b2)}{area(b1 \cup b2)}$.

Occlusion type features: Since the above process may miss some occlusion information due to unannotated objects, we also use an occlusion index. The index represents whether an object is not occluded, partially occluded, heavily occluded, or includes an unknown occlusion type.

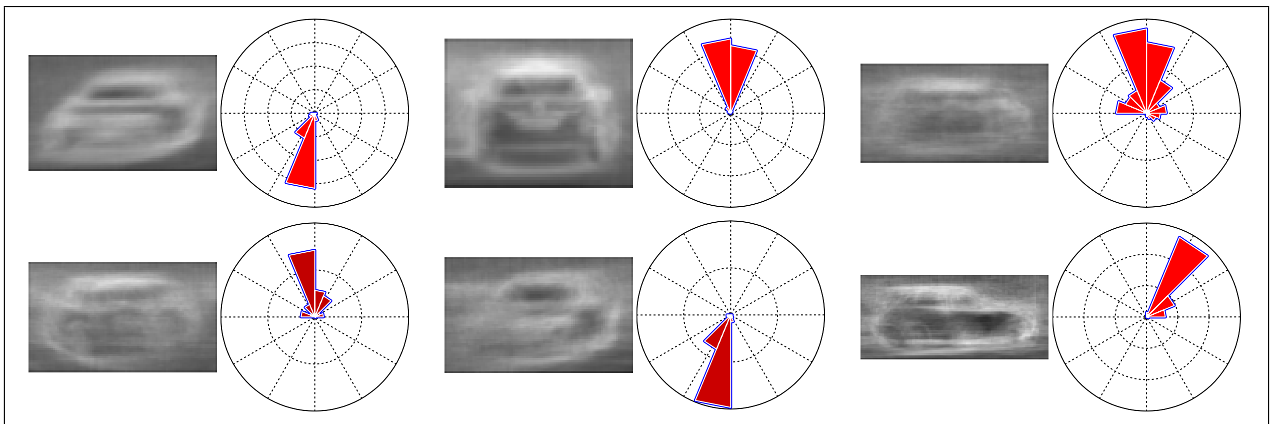
As occluded objects are common in the KITTI dataset, a set of features is extracted to further refine the occlusion types. The relative orientation, $\theta_{occludee} - \theta_{occluder}$ provides additional context for the type of the occlusion. For instance, as the occluder's orientation varies, so does the appearance of the occluded object in the window patch. Therefore, the occluder orientation, $\theta_{occluder}$, as well as the relative 3D position, $\mathbf{p}_{occludee} - \mathbf{p}_{occluder}$ ($\mathbf{p} \in \mathbb{R}^3$), are used as features as well. Finally, a binary feature is used to encode whether the occluder is on the left or the right of the occludee using the centroid coordinates of the bounding boxes.

C. Clustering

Fig. 3 shows the color and gradient features in a two-dimensional representation. Each point is binned into a color according to its orientation and occlusion level (either occluded or non-occluded). We observe how clusters are not well separated, demonstrating how visual subcategorization is a difficult problem. On the other hand, the 2D projection



(a) Discriminative subcategorization (**DSC** [6]) with only **visual features**.



(b) Clustering with k-means on the proposed set of **geometrical features**.

Fig. 5: Clustering to 6 clusters with two types of features. (a) Visual features (color, gradient magnitude, and gradient histogram features at 6 bins) vs. (b) k-means on geometrical features. Rose plots show the orientation distribution of the samples in each centroid. Color shows a percentage of occluded samples, varying from light red (no occlusion) to dark (occlusion). Note how k-means with geometrical features produces distinct clusters both in terms of orientation (as opposed to DSC on visual) and in terms of occlusion level.

using the set of geometrical features of the same data produces well separated clusters. The labels were kept the same over the experiments in visual and geometrical features. Intuitively, this representation is correlated with how well an object detector would perform (using geometrical features generally outperforms visual feature clustering in this study).

Nonetheless, quantization of the feature space is still not trivial. For instance, certain geometrical variations may not correspond to significant appearance variation (hence such variations can be included in the same model). One possible quantization can be performed in a unsupervised or semi-supervised fashion using a clustering algorithm, as shown in Fig. 3(c).

Strategy 1 for orientation and occlusion modeling: A uniform binning of orientation bins and occlusion level bins. Since the algorithm described in Section III-B for extracting the occlusion level may miss some occluded instances (for instance, if a pole or another non-vehicle object was the occluder), we also incorporate the occlusion index when binning (so that occluded cases will not fall into clusters containing non occluded vehicles). Even still, there are several possible quantization techniques.

First, occluded and partially-occluded vehicles may be grouped together into the same cluster, so that M (the occlusion quantization parameter in Fig. 4) is set to 1 and B is varied.

Second, we may entirely split occluded and not occluded cases in all of the analysis, referred to as **Split** in the experimental analysis in Section VII.

Third, we may vary both B and M. For instance, if the maximum occlusion level in all of the samples is 80%, a value of $M = 2$ would create a quantization over $[0 - 40\%]$ and $[41\% - 80\%]$ for each orientation bin.

Strategy 2 for orientation and occlusion modeling: To account for all the variables that influence appearance variations (such as occluder statistics, truncation, etc. see Section III-B), we may cluster over those features directly, using k-means or spectral clustering **SC**, which we found to work well.

Strategy 3 for orientation and occlusion modeling: In this strategy, 3D geometry features are used in order to initialize a visual subcategorization routine, such as LSVM or the framework in [6]. Generally, this produced minor improvement on the final detection results in our experiments.

Strategy 4 for orientation and occlusion modeling: Clustering of visual features only, as described in III-A.

Comments on unsupervised or weakly-supervised clustering methods: k-means and spectral clustering will be used. In our implementation of spectral clustering, a Gaussian kernel is employed as a similarity function between two samples $\mathbf{x}_i$ and $\mathbf{x}_j$, $W_{ij} = \exp \frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{2\sigma^2}$. We then compute the normalized graph Laplacian, $L = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$, where D is the diagonal degree matrix. Next, k-means is run on the L_2 normalized matrix of eigenvectors of L. These two clustering techniques will be compared against the discriminative subcategorization framework of [6] (referred to as **DSC**) both with visual and geometrical features. DSC employs a weakly-supervised framework for obtaining cluster labels with the presence of negative samples. As in [6], our experiments showed DSC to be superior to Latent SVM in prevention of degenerate clusters and overall cluster purity. DSC utilizes a block coordinate gradient-descent alternating between optimization of the SVM parameters and the cluster labels. Different initialization schemes for DSC will be studied. Generally, we found that the procedure of first training a linear SVM on the positive and negative instances to obtain a weight vector $\mathbf{w}$, and clustering the residual vectors after projection on $\mathbf{w}$ using $\mathbf{x} - \frac{1}{||\mathbf{w}||}(\mathbf{w}^T\mathbf{x})\mathbf{w}$ improved the final clustering quality. For negative samples, we use three iterations of hard negative mining. Fusion of the two types of features, visual and geometric, is also of interest, as a partition in the geometrical space may not be correlated with a partition in the visual space.

IV. Detection Framework

AdaBoost [7] is learned using depth-2 decision trees as weak classifiers. Detection at multiple scales is handled using approximation of features at nearby scales, as in [32]. The color and gradient image features are aggregated in 4×4 blocks in order to produce fast pixel lookup features.

Training parameters: In all of the experiments, training a component involves hard mining of negative instances was performed, with the first stage sampling 5000 random negative samples, followed by three additional stages of training using hard negatives. In each round, instances belonging to the other subcategories are excluded as negatives. This exclusion takes place according to an overlap threshold with the detection. In the range from $\{0.1, 0.15, 0.2, \ldots, 0.5\}$, 0.35 was shown to work best for exclusion ([33] also found 0.3 to work well on the PASCAL dataset).

Pooling detectors: Given a test image, the trained models for each subcategory are all evaluated. Overlapping detections are merged using a greedy non-maximum suppression (NMS) procedure; once a bounding box is suppressed by the overlap criterion, it can no longer suppress weaker detections. We experimented with two NMS schemes: one using the PASCAL overlap criteria of intersection-over-union (defined in Section III-B), and a second scheme where the union denominator is replaced by the minimum area of the two bounding boxes. Best results were shown with the first NMS scheme. We did not find it necessary to carefully calibrate the models as in [3], as the gains were not significant once the best choice of features and clustering technique was made.

V. Orientation Estimation

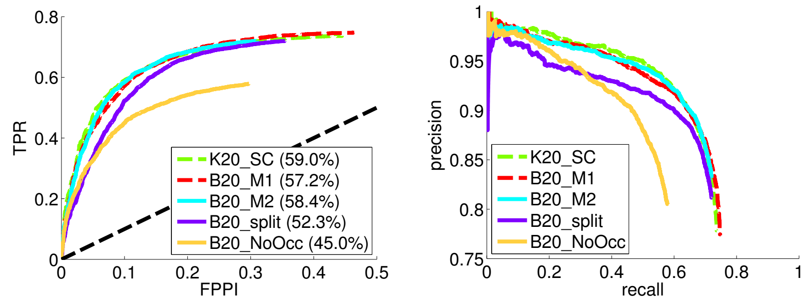
Due to the close relationship between the clustering and vehicle orientation, this immediately motivates the study of orientation estimation. In particular, we care about the relationship between the number of subcategories and orientation estimation accuracy. Furthermore, the impact that different occlusion handling strategies have on orientation estimation is also of interest.

For detection, we generally did not find a need for carefully calibrating the scores outputted by each model. Normalization of the output of each detector to the $[0, 1]$ range linearly or using a sigmoid provided a negligible improvement. Nonetheless, several issues may arise with orientation estimation. For instance, it was observed that detectors at about π difference in orientation would both fire together. For instance, rear and front instances would sometimes get mixed, as well as left and right orientations. This is intuitive, but requires a more careful analysis of the scores output. Therefore, two approaches were considered for performing the final orientation estimation, one is using classification and one using regression. For regression, we use a L2-regularized L2-loss support vector regression [34]. For classification, we use a Crammer and Singer multiclass SVM [35]. In the latter, a weight w is learned for each class, and these weights are optimized as a whole. Both are used with a linear kernel.

With a large number of clusters, it is expected that multiple detectors at nearby orientations (or possibly at opposite orientations) would fire. In order to compare among all clustering methods, a mapping is learned from the detectors' scores at sufficient spatial proximity to a 3D orientation value. Given the set of detections in an image, D, we construct a feature vector for each detection box as following. Each detection is defined by a bounding box B, a score s, and the associated model k, so that $(B, s, k) \in D$. First, NMS is performed in order to produce sparse detection boxes and fixing detection performance to the one in Section IV. Consequently, for leveraging context in the final orientation estimate, NMS is performed on each detector individually and a feature vector is constructed using the maximum score of each detector that has a higher overlap than 0.5 with the given post-NMS detection. Therefore, k models produce a k dimensional feature vector of scores, $(s_1, ..., s_K)$. When compared to using supervised orientation clusters and taking the orientation corresponding to the maximally scored detector, the SVM approach was found to slightly improve estimate accuracy.

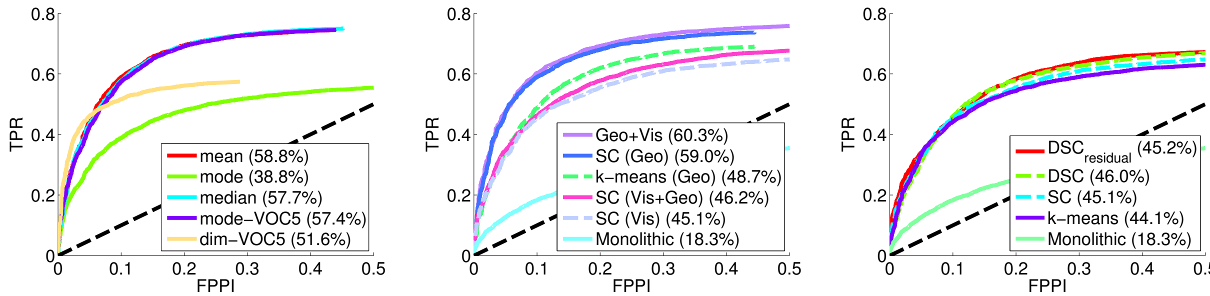
In training, the models are evaluated on the annotated training images, so that each true positive contributes a training sample for the orientation estimation model. The scores are linearly normalized before inputting to the SVM.

Orientation estimation is evaluated using the orientation



(a)

Fig. 6: Strategy 1: Varying a fixed bin parameter in orientation space (**B**) and occlusion level space (**M**). For $M > 2$, no improvement was gained. The results are compared with Strategy 2, spectral clustering on geometrical features ($K20$_SC or **SC (Geo)**) which is the best performing.



(a) (b) (c)

Fig. 7: Results for $K = 20$ subcategories. (a) Given a clustering assignment, how should model dimensions be determined? (b) Analysis of the clustering techniques and different features (strategies 2 and 4), see Section VII for more detail. Spectral clustering (**SC**) with geometrical (**Geo**) features is shown to work well, better than purely visual (**Vis**) subcategorization. (c) Strategy 4; Clustering analysis for only visual features defined in Section III.

similarity metric proposed in [29],

$$s(r) = \frac{1}{|D(r)|} \sum_{i \in D(r)} \frac{1 + cos\Delta_\theta^i}{2} \delta_i \tag{3}$$

where, for a given recall rate r, $D(r)$ is the set of object detections and Δ_i is the angle difference between estimated and ground truth orientation. δ_i is set to 1 if detection i has been assigned to a ground truth bounding box and 0 otherwise.

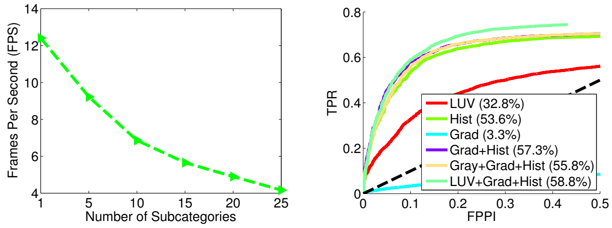
VI. Experimental Settings

For evaluation of the proposed framework, the KITTI dataset is used [29]. Three evaluation methods were suggested in [29], 'easy', 'moderate', and 'hard' with increasing occlusion and truncation and decreasing minimum object size. There are 7481 training images (with over 20,000 vehicle instances), which were split in half to produce a training and a validation dataset. All the experiments employed a 70% overlap requirement in order for a detection to count as a true positive, and are performed by testing with 'moderate' test settings.

VII. Experimental Evaluation

For clarification, it is pointed out that throughout the experiments the letter B refers to the number of uniform 3D orientation bins, M refers to occlusion level bins, and K is used for the unsupervised or weakly-supervised clustering experiments such as spectral clustering (**SC**). $K20\_SC$ refers to strategy 2, where the geometrical features are clustered to 20 components, and is the same as **SC (Geo)**.

Model parameters: The results for this analysis are shown in Fig. 7(a). In parenthesis for all curves and plots, the detection rate at 10^{-1} FPPI rate is shown. For obtaining the model dimensions of each cluster, several options were considered. One may determine an aspect ratio for each cluster using the mean, mode, or median of the samples' aspect ratios. In these approaches, one of the dimensions is always kept fixed, and the other is derived from the aspect ratio. Careful optimization showed fixing one dimension at 32 pixels worked best (used in all of the experiments). The DPM-VOC version 5 code [1] is commonly used in object detection studies, yet two approaches based on it sub-optimal. In **mode-VOC5**, the aspect ratios of the samples in each cluster were filtered as in the available implementation in the process of obtaining the mode. Then, a base dimension of 32 was used to obtain the dimensions of each component. In **dim-VOC5**, the entire pipeline from [1] was used, which determines model dimensions by picking the 20th percentile area (as opposed to fixing one dimension at 32). Note that a monolithic classifier runs at ~12.5 fps on a CPU on full resolution images of size 1242 × 375. Model padding was also grid optimized, and 1/8 of the model size



(a) (b)

Fig. 8: (a) Impact of varying the number of subcategories on frame-rate of the entire detection pipeline. (b) Impact of the feature types on detection performance.

in each dimension (width and height) was used for padding.

Object subcategorization strategies: Here we study clustering and feature combinations. First, strategy 1 is evaluated in Fig. 6, where it is shown that a good approach is simply to set $M = 1$. We note that this means that in training for a specific orientation, we keep both occluded and non-occluded samples in the same cluster. Little benefit was made by learning occlusion separate models as shown for $M = 2$. This is unlike the study of [4], possibly due to the different modeling technique. For instance, our approach do not explicitly model a notion of parts, which could be useful for learning occlusion-only subcategory models. For $M = 2$, 40 models are required to be learned at $B = 2$. Performing **split**, where fully visible samples are separated from samples with any kind of occlusion, also under-performed the $M = 1$ binning. Furthermore, we also observed a slight improvement using strategy 2 and spectral clustering (**SC**), which performed best among all strategies for a fixed size of $K = 20$ models.

The remaining strategies are analyzed in Fig. 7. Fig. 7(c) shows an analysis for purely visual subcategorization, where DSC is shown to produce better detection results as opposed to spectral clustering or just k-means. Strategy 3, where DSC is initialized using ground truth 3D information or occlusion levels, resulted in minor improvements to the $\text{DSC}_{\text{residual}}$. Furthermore, although in the original implementation [6] k-means is used for initialization to DSC, using SC for initialization was shown to improve results as opposed to k-means. Therefore, all the results for DSC are shown with SC initialization.

Fusion of geometric and visual features: As mentioned, DSC initialization using geometry features did not significantly improve the final detection performance. As a matter of fact, even incorporation of visual features with geometrical ones in the clustering degraded results over using geometrical features only. Two approaches did provide a slight improvement using fusion. Firstly, in the computation of the affinity matrix W, the features were decomposed into a weighted combination over the visual and geometrical Euclidean similarities. This allowed us to weigh geometrical features more before clustering was performed. A slight improvement was noted. Secondly, learning a separate set of $K = 20$ subcategories and running all 40 models resulted in an improvement as well (shown as **Geo+Vis** in Fig. 7(b)). The gains were small, hence SC and geometrical features are used in most of the analysis.

Number of subcategories: As shown in Fig. 9, $K = 20$

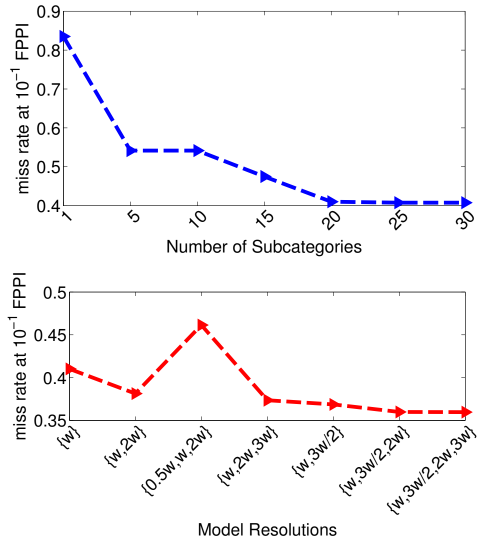


Fig. 9: Top: Results of varying K, the number of subcategories. $K = 20$ is shown to work well. Bottom: Incorporating multiple resolutions provides additional benefits. $w = 32$ in the experiments.

provided a good choice. Even with 20 models, the method detects at ∼5 fps on a CPU on full resolution images.

Conclusion for subcategorization: Either strategy 1 with $M = 1$ or strategy 3 with SC work well. It appears that the most important variation in vehicle detection to account for is due to orientation (also clearly visible in Fig. 10(d)). Improving occlusion handling was best done by strategy 1 or by utilizing an unsupervised approach as in strategy 3.

Resolution components: Appearance also varies due to the scale and distance of the object from the camera. As seen in Fig. 9, adding detectors trained on varying resolutions was shown to impacts performance significantly (the work of [8] reported mild improvements for pedestrian detection). A main reason for this is improved localization of detected objects, as we noticed how higher scores would be produced by the model resolution most appropriate for the object's image dimensions. Employing different resolution models is another way of capturing variation in object appearance among scales. Out of the different possibilities of scales, we found learning models at $w = 32$, $w = 40$, and $w = 48$, corresponding to scales 1.25 and 1.5 respectively, produced the most significant performance jump. Although fps rate decreases with each scale addition, at this stage of the framework we were mostly concerned with obtaining the optimal performance, as the entire detection pipeline is still significantly faster than the DPM (even with multiresolution models). Nonetheless, there many possible speedups which could explore the redundancy among the models and detection at different scales. These are left for future work.

Application to pedestrian detection: We also show that the proposed framework can be applied for other domains of object detection. For pedestrian detection, we follow the same framework, but we noticed less obvious advantages between subcategorization with different features. Intuitively, a model

TABLE II: Evaluation on the KITTI testing benchmark. (a) Area under the precision-recall curve with varying test settings for the detection task. The top three methods are shown in bold. (b) Area under the orientation similarity-recall curve for detection and orientation estimation curve. DPM-based methods employ version-4 of the available implementation unless stated as **V5**.

(a) Car detection

Method	**Easy** (%)	**Moderate** (%)	**Hard** (%)
Regionlets [36], [37]	**84.27**	**75.58**	**59.20**
SubCat (Ours)	**81.94**	**66.32**	51.10
AOG [16]	**80.26**	**67.03**	**55.60**
SpCov_ACF [38]	78.67	58.19	44.80
DPM (V5) [16]	77.24	56.02	43.14
SpCov [38]	76.53	62.29	48.00
OC-DPM [4]	74.94	65.95	**53.86**
DPM-C8B1 [39]	74.33	60.99	47.16
LSVM-MDPM-sv [1]	68.02	56.48	44.18
LSVM-MDPM-us [1]	66.53	55.42	41.04
ACF [7]	55.89	54.74	42.98
mBoW [40]	36.02	23.76	18.44

(a) Car detection and orientation estimation

Method	**Easy** (%)	**Moderate** (%)	**Hard** (%)
SubCat (Ours)	**80.92**	**64.94**	50.03
OC-DPM [4]	73.50	64.42	**52.40**
LSVM-MDPM-sv [1]	67.27	55.77	43.59
DPM-C8B1 [39]	59.51	50.32	39.22
AOG [16]	44.41	36.87	30.29

learned for frontal pedestrians may leverage face color cues, but a model learned for pedestrians viewed at rear may weight a different set of features. This is an example of how the framework studied can be generalized to improve general object detection, important for other domains of intelligent transportation systems[41]–[43]. For pedestrians, the height of the bounding box is fixed at $h = 64$. For padding, we use $1/4$ model height size in the vertical direction (so total padding is $1/2$ model height size), and half the width size on the left and right of each pedestrian sample. Results are shown in Fig. 11. For this evaluation, an overlap threshold of 50% is used.

Orientation estimation results: As shown in Fig. 10(a), the multiclass SVM produced significantly better orientation estimation compared to support vector regression which outputs a continuous value. The number of orientation bins is analyzed in Fig. 10(b), and we see a plateau after $B = 25$. For these experiments, we set $M = 1$ as it appears little value was gained for orientation estimation by incorporating different occlusion level models. This is demonstrated in Fig. 10(d), where the results are even more clearly observable once multiresolution models are incorporated.

Comparison with state-of-the-art: Table II shows the detection and orientation estimation results on the KITTI testing dataset. We also refer the reader to the online evaluation board at http://www.cvlibs.net/datasets/kitti. Since the detection task contains many entries, the three top methods in each evaluation category are bolded. Interestingly, some techniques perform well on detection (AOG) but poorly estimate orientation. Because the different approaches employ different computational environments, no explicit comparison in terms of speed can be made. Nonetheless, our submission provides a speedup of about a factor of 10-30 over reported speeds of varying DPM-based approaches. Remarkably, the fastest techniques which employ the same baseline detection framework as ours (ACF) do not nearly perform at the same level, even with richer features [38]. Our approach is shown to significantly improve detection performance over and out-of-the-box ACF-based submission by a large margin of 26%, 11%, and 8% in 'easy', 'moderate', and 'hard' test settings, respectively. This in turn brings the ACF approach from one of the lowest-performing approach to one of the top-performing. The current state-of-the-art is achieved by the Regionlets approach [36], which relies on an elaborate set of features: HOG, Local Binary Patterns (LBP), Covariance, and Convolutional Neural Network (CNN) features. The improvement from adding such features is orthogonal to our approach, which only utilizes HOG+LUV features. The approach is also slower than ours, by about a factor of 3. Although we opted to using feature approximation in detection over scales and downsampling of the HOG+LUV pixel lookup features by a factor of 4, this degrades performance (but it provides fast training and testing). For instance, by downsampling only by a factor of 2 instead of 4, not approximating features in test time, and allowing for mining more negatives we were able to suppress the Regionlets approach in performance across evaluation methods. As these improvements are orthogonal to the studied approach, such tweaks will be associated and further studied in a future work.

VIII. Concluding Remarks

In this paper, the role of object subcategories was studied for detection at multiple geometrical configurations and occlusion, focusing on vehicle detection. Appearance variations were encoded by learning a specific model for vehicle instances at certain occlusion types or orientations. Using fast feature extraction and detection schemes, we were able to achieve excellent detection performance while comparing favorably in terms of run-time to other common approaches.

Visual categorization could be further studied in the future, such as using CNN features (e.g. Caffe [33], [45]). Although the results are promising, future work would focus on improving occluded vehicle detection. For instance, heavily occluded vehicles (especially in parked settings at side-view) are commonly missed among all state-of-the-art techniques for vehicle detection. In our analysis, the largest gain came from quantizing over orientation bins, and learning subcategory clusters with both moderately occluded and little occluded samples (unlike other studies, such as [46] which completely exclude occluded samples). Furthermore, we observed a large drop in performance when using the more strict 70% overlap evaluation threshold as opposed to the common 50% (also shown in [39]), indicating better localization is required. This could be addressed using regression approaches, as in [36], [47]. Truncated settings are challenging as well, as shown in

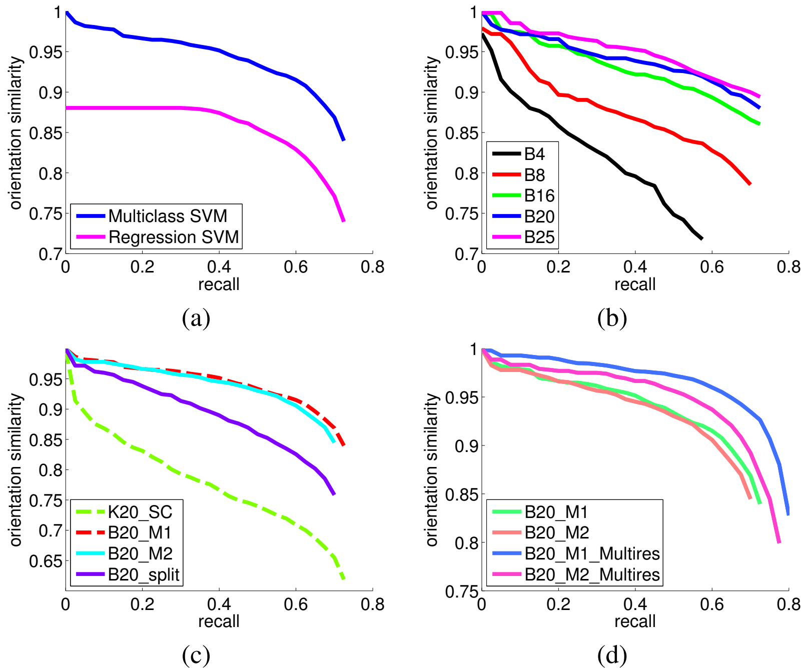


Fig. 10: Analysis for orientation estimation of vehicles. (a) Classification with a multiclass SVM vs. support vector regression. (b) Impact of number of orientation bins (at occlusion quantization $M = 1$) on orientation similarity results. (c) Different occlusion handling techniques and their effect on orientation estimation. Although K20_SC provided the best detection results, the unsupervised framework comes at a cost for orientation estimation. (d) The effects of multiresolution model learning on orientation estimation. We note how here, employing $M = 1$ performs significantly better over further occlusion quantization.

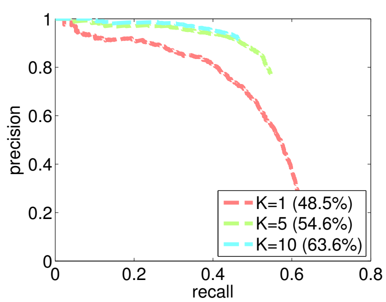


Fig. 11: Using the proposed framework to learn subcategory models for pedestrian detection favorably impacts performance over a monolithic classifier ($K = 1$).

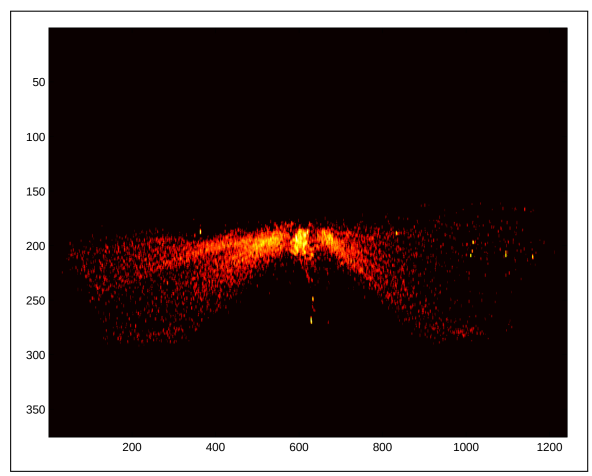


Fig. 12: Center location distribution (log normalized) obtained from ground truth. Scene information can be leveraged for improved detection performance and possible speedups, as in [44]. This is left for future work.

Fig. 13. Further improvements can be made by incorporating motion features as in [48]. Further speedups and performance improvement can be gained by incorporating scene information for rescoring detections [44], [49] or the fast detection approach of [9].

REFERENCES

[1] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[2] E. Ohn-Bar and M. M. Trivedi, "Fast and robust object detection using visual subcategories," in *Computer Vision and Pattern Recognition Workshops-Mobile Vision*, 2014.

[3] S. K. Divvala, A. A. Efros, and M. Hebert, "How important are deformable parts in the deformable parts model?" in *European Conf. Computer Vision Workshops*, 2012.

[4] B. Pepik, M. Stark, P. Gehler, and B. Schiele, "Occlusion patterns for object class detection," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2013.

[5] R. Girshick, F. Iandola, T. Darrell, and J. Malik, "Deformable part models are convolutional neural networks," *CoRR*, 2014.

[6] M. Hoai and A. Zisserman, "Discriminative sub-categorization," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2013.

[7] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2014.

[8] R. Benenson, M. Mathias, R. Timofte, and L. V. Gool, "Pedestrian

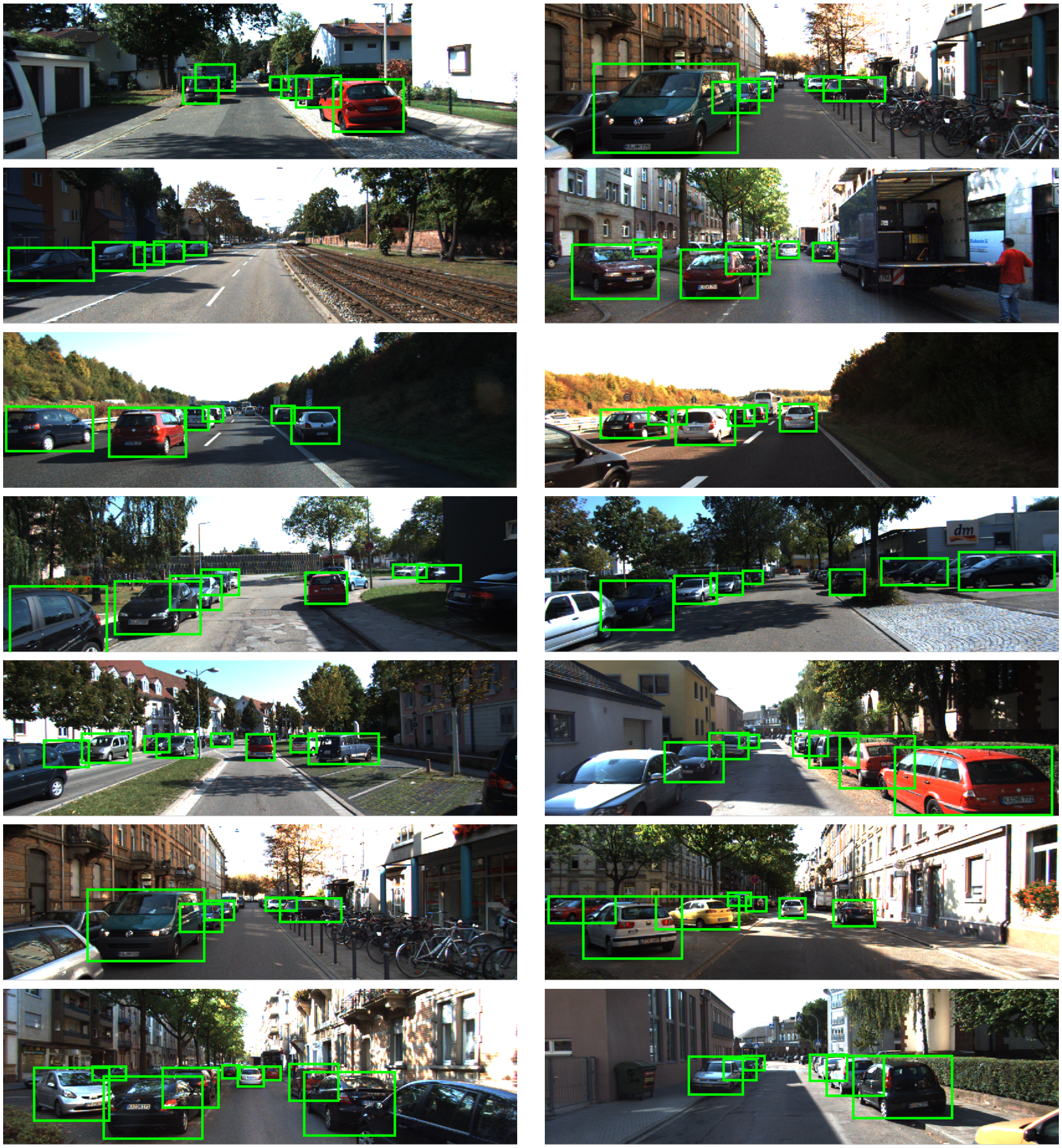


Fig. 13: Detection results on the KITTI dataset. A strength of the system is robustness against false positives. Future work should improve occlusion handling, localization tightness, and truncated vehicle detection. Note that trucks are considered a different class from the car class and are not used for training or evaluation.

detection at 100 frames per second," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2012.

[9] T. Dean, M. Ruzon, M. Segal, J. Shlens, S. Vijayanarasimhan, and J. Yagnik, "Fast, accurate detection of 100,000 object classes on a single machine," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2013.

[10] M. A. Sadeghi and D. Forsyth, "30Hz object detection with DPM V5," in *European Conf. Computer Vision*, 2014.

[11] B. T. Morris and M. M. Trivedi, "Trajectory learning for activity understanding: Unsupervised, multilevel, and long-term adaptive approach," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2011.

[12] C.-H. Kuo and R. Nevatia, "Robust multi-view car detection using unsupervised sub-categorization," in *IEEE Winter Conf. Applications of Computer Vision*, 2009.

[13] H. T. Niknejad, A. Takeuchi, S. Mita, and D. McAllester, "On-road multivehicle tracking using deformable object model and particle filter with improved likelihood estimation," *IEEE Trans. Intelligent Transpotation Systems*, vol. 13, no. 2, pp. 748–758, 2012.

[14] M. Hejrati and D. Ramanan, "Analyzing 3D objects in cluttered images," in *Advances in Neural Information Processing Systems*, 2012.

[15] B. Li, W. Hu, T. Wu, and S.-C. Zhu, "Modeling occlusion by discriminative and-or structures," in *IEEE Intl. Conf. Computer Vision*, 2011.

[16] B. Li, T. Wu, and S.-C. Zhu, "Integrating context and occlusion for car detection by hierarchical and-or model," in *European Conf. Computer Vision*, 2014.

[17] S. Sivaraman and M. M. Trivedi, "Vehicle detection by independent parts for urban driver assistance," *IEEE Trans. Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1597–1608, 2013.

[18] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2005.

[19] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A

Fig. 14: Orientation estimation results on the KITTI dataset. In red is the estimated orientation and in blue the ground truth orientation.

survey of vision-based vehicle detection, tracking and behavior analysis," *IEEE Trans. Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1773–1795, 2013.

[20] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun, "3D traffic scene understanding from movable platforms," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2014.

[21] H. T. Niknejad, T. Kawano, Y. Oishi, and S. Mita, "Occlusion handling using discriminative model of trained part templates and conditional random field," in *IEEE Intelligent Vehicles Symposium*, 2013.

[22] B. Li, W. Hu, T. Wu, and S.-C. Zhu, "Modeling occlusion by discriminative and-or structures," in *IEEE Intl. Conf. Computer Vision*, 2013.

[23] S. Tang, M. Andriluka, and B. Schiele, "Detection and tracking of occluded people," *Intl. Journal of Computer Vision (to appear)*, 2014.

[24] S. Tang, M. Andriluka, A. Milan, K. Schindler, S. Roth, and B. Schiele, "Learning people detectors for tracking in crowded scenes," in *IEEE Intl. Conf. Computer Vision*, 2013.

[25] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems*, 2001.

[26] T. Lan, M. Raptis, L. Sigal, and G. Mori, "From subcategories to visual composites: A multi-level framework for object detection," in *IEEE Intl. Conf. Computer Vision*, 2013.

[27] X. Zhu, C. Vondrick, D. Ramanan, and C. C. Fowlkes, "Do we need more training data or better models for object detection?" in *British Machine Vision Conf.*, 2012.

[28] C. Gu and X. Ren, "Discriminative mixture-of-templates for viewpoint classification," in *European Conf. Computer Vision*, 2010.

[29] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[30] L. V. der Maaten and G. Hinton, "Visualizing data using t-SNE," *JMLR*, vol. 9, no. 85, pp. 2579–2605, 2008.

[31] A. Geiger, F. Moosmann, O. Car, and B. Schuster, "A toolbox for automatic calibration of range and camera sensors using a single shot," in *IEEE Conf. Robotics and Automation*, 2012.

[32] P. Dollár, S. Belongie, and P. Perona, "The fastest pedestrian detector in the west," in *British Machine Vision Conf.*, 2010.

[33] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2014.

[34] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[35] K. Crammer and Y. Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *Journal of Machine Learning Research*, vol. 2, pp. 265–292, 2001.

[36] C. Long, X. Wang, G. Hua, M. Yang, and Y. Lin, "Accurate object detection with location relaxation and regionlets relocalization," in *Asian Conf. on Computer Vision*, 2014.

[37] X. Wang, M. Yang, S. Zhu, and Y. Lin, "Regionlets for generic object detection," in *IEEE Intl. Conf. Computer Vision*, 2013.

[38] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, "Strengthening the effectiveness of pedestrian detection with spatially pooled features," in *European Conf. Computer Vision*, 2014.

[39] J. J. Yebes, L. M. Bergasa, R. Arroyo, and A. Lázaro, "Supervised learning and evaluation of KITTIs cars detector with dpm," *Intelligent Vehicles Symposium*, 2014.

[40] J. Behley, V. Steinhage, and A. B. Cremers, "Laser-based Segment Classification Using a Mixture of Bag-of-Words," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[41] E. Ohn-Bar, S. Martin, and M. M. Trivedi, "Driver hand activity analysis in naturalistic driving studies: Issues, algorithms and experimental studies," *Journal of Electronic Imaging*, vol. 22, pp. 1–10, 2013.

[42] T. Gandhi and M. M. Trivedi, "Pedestrian protection systems: Issues, survey, and challenges," *IEEE Trans. Intelligent Transportation Systems*, vol. 8, pp. 413–, 2007.

[43] E. Ohn-Bar and M. M. Trivedi, "In-vehicle hand activity recognition using integration of regions," in *IEEE Intelligent Vehicles Symposium*, 2013.

[44] S. Sivaraman and M. M. Trivedi, "Integrated lane and vehicle detection, localization, and tracking: A synergistic approach," *IEEE Trans. Intelligent Transportation Systems*, vol. 14, no. 2, pp. 906–917, 2013.

[45] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[46] D. Park, C. Zitnick, D. Ramanan, and P. Dollár, "Exploring weak stabilization for motion feature extraction," *IEEE Conf. Computer Vision and Pattern Recognition*, 2014.

[47] S. Schulter, C. Leistner, P. Wohlhart, P. M. Roth, and H. Bischof, "Accurate object detection with joint classication-regression random forests," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2014.

[48] A. Ramirez, E. Ohn-Bar, and M. M. Trivedi, "Integrating motion and appearance for overtaking vehicle detection," *IEEE Intelligent Vehicles Symposium*, 2014.

[49] K. Matzen and N. Snavely, "NYC3DCars: A dataset of 3D vehicles in geographic context," in *IEEE Intl. Conf. Computer Vision*, 2013.

PLACE PHOTO HERE

Eshed Ohn-Bar is currently working towards a Ph.D. degree in electrical engineering with specialization in signal and image processing at the Computer Vision and Robotics Research Laboratory and LISA: Laboratory for Intelligent and Safe Automobiles at the University of California, San Diego.

PLACE PHOTO HERE

Mohan Manubhai Trivedi is a Professor of electrical and computer engineering and the founding director of the Computer Vision and Robotics Research Laboratory and Laboratory for Intelligent and Safe Automobiles (LISA) at the University of California, San Diego. He and his team are currently pursuing research in machine and human perception, machine learning, human-centered multimodal interfaces, intelligent transportation, driver assistance and active safety systems. Trivedi serves as a consultant to industry and government agencies in the U.S. and abroad, including the National Academies, major auto manufactures and research initiatives in Asia and Europe. Trivedi is a Fellow of the IEEE (for contributions to Intelligent Transportation Systems field), Fellow of the IAPR (for contributions to vision systems for situational awareness and human-centered vehicle safety), and Fellow of the SPIE (for distinguished contributions to the field of optical engineering).