2012 15th International IEEE Conference on Intelligent Transportation Systems
Anchorage, Alaska, USA, September 16-19, 2012

Real-Time Vehicle Detection Using Parts at Intersections

Sayanan Sivaraman and Mohan M. Trivedi

***Abstract*—In this study, we propose a novel, lightweight approach to real-time detection of vehicles using parts at intersections. Intersections feature oncoming, preceding, and cross traffic, which presents challenges for vision-based vehicle detection. Ubiquitous partial occlusions further complicate the vehicle detection task, and occur when vehicles enter and leave the camera's field of view. To confront these issues, we independently detect vehicle parts using strong classifiers trained with active learning. We match part responses using a learned matching classification. The learning process for part configurations leverages user input regarding full vehicle configurations. Part configurations are evaluated using Support Vector Machine classification. We present a comparison of detection results using geometric image features and appearance-based features. The full vehicle detection by parts has been evaluated on real-world data, runs in real time, and shows promise for future work in urban driver assistance.**

***Index Terms* - Active Safety, Driver Assistance, Real-time Vision, Machine Learning.**

I. INTRODUCTION

According to NHTSA, in 2009, automotive collisions in urban environments accounted for 43% of fatal crashes in the United States. Tens of thousands of peoples are killed on the roads each year, and most fatal crashes feature more than one vehicle [1]. As research in sensing and environmental perception progresses, there is great potential to save lives by developing advanced driver assistance systems. Over the past decade, there has been significant research effort dedicated to the development of intelligent driver assistance systems, intended to enhance safety by monitoring the driver and the on-road environment [2]. In particular, surround analysis and understanding using vehicle-based sensing will be crucial to enhancing the safety of drivers, vehicle occupants, and other road users.

This work deals with on-road detection of vehicles in urban environments. Vehicle detection and surround analysis using computer vision presents many challenges [3], with urban environments especially so. When approaching urban intersections, vehicles are encountered exhibiting a variety of orientations [4]. In particular, vehicles comprising oncoming traffic, preceding traffic, and cross traffic are all encountered. Further, cross traffic is often subject to partial occlusions, especially upon entry and exit from the camera's field of view. The ideal vehicle detection system needs to handle all these cases robustly and efficiently to reliably enhance on-road safety.

In this study, we propose a lightweight approach to vehicle detection at intersections, using a combination of parts, intended to deal with the challenges of the urban environment. Vehicle parts are individually detected using strong classifiers, trained using active learning [5]. An interactive interface has been developed, which allows a user to label which parts comprise a given vehicle in a test frame. The interactive matching data is used to train a Support Vector Machine classifier for vehicle part matching [6].

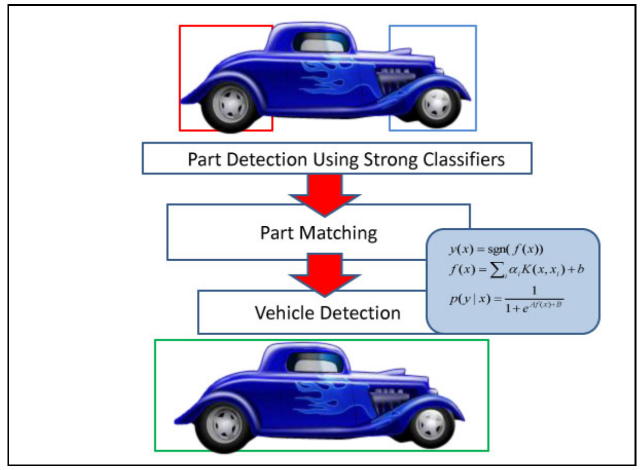

Fig. 1. Proposed approach to vehicle detection by parts. Individual parts of vehicles, the front and rear parts, are detected independently using strong classifiers, trained using active learning [5]. Parts are then matched using Support Vector Machines. Finally, vehicles are detected as a combination of parts. Best viewed in color.

Figure 1 depicts the overall approach. On a given frame, classifiers are applied to detect the front parts, and rear parts of vehicles. Given the set of detected parts, matching is implemented to form full vehicles using a trained SVM classifier. Matched parts are used to form side-profile vehicles, while unmatched parts are retained as oncoming, preceding, or partially occluded vehicles.

The main contributions include the following. Using active learning, vehicle parts are independently detected in a given frame. Using a trained classifier, individual vehicle parts are matched to form and detect full vehicles. Using a combination of independent part detectors allows the proposed system to detect oncoming vehicles, preceding vehicles, cross traffic vehicles, and to handle partial occlusions. We have compared the detection performance using both geometric, and appearance-based image features for matching independently-detection vehicle parts. Using the primal form of the linear SVM, the full system runs in real time, with no specific hardware or software optimizations.

The remainder of this paper consists of the following. Section 2 briefly describes related research. Section 3 describes overall approach to vehicle detection by parts. Section 4 provides experimental evaluation. Finally, Section 5 offers discussion and concluding remarks.

II. Related Research

Robust detection of other vehicles on the road using vision is a challenging problem. Roads are dynamic environments, with ever-changing backgrounds and illuminations. The ego vehicle and the other vehicles on the road are generally in motion. The sizes and locations of vehicles in the image plane are diverse, although they can be modeled [7]. There is high variability in the shape, size, color, and appearance of vehicles found in typical driving scenarios [8]. The desire for systems that can localize vehicles in real-time imposes additional constraints on the development of vehicle detection systems.

Many works in vehicle detection over the past decade have focused on detection of the rear face of vehicles [8], [5], [9], [7]. Detectors of this sort are designed for preceding, or sometimes oncoming traffic. In [7], feature points are tracked over a long period of time, and vehicles detected based on the tracked feature points. The distribution of vehicles in the image plane is modeled probabilistically, and a hidden Markov model formulation is used to detect vehicles and separate them from the background.

Detecting objects by parts has been explored in the computer vision community in various incarnations. Much attention has been focused on the detection of pedestrians by parts, as a way of handling partial occlusions, and reducing false alarm rates. In [10], individual parts are detected using strong classifiers, and pedestrians are constructed using a Bayesian combination of parts. The number of detection pedestrians, and their locations, are the most likely part-based configuration. In [11], a more efficient feature representation is used, and parts are chosen to be semantically meaningful and overlapping in the image plane.

In [12], multiple instance learning is used for part-based pedestrian detection. The study demonstrates how the ability of multiple instance feature learning to deal with training set misalignment, can enhance performance of part-based object detectors. In [13], pedestrian parts are manually assigned to semantically meaningful parts, their physical configuration manually constrained and overlapping. The combination of parts is a weighted sum, with higher weights going to more-reliably detected parts. In [14], this work is extended, with further experiments on the feature set.

The work of [15] for object detection using a deformable, parts-based model, based on the Latent Support Vector Machine has introduced new avenues for detection of vehicles by parts. While [15] demonstrates vehicle detection evaluated on static images, the object detection framework is used for video-based nighttime vehicle detection in [16]. Integrated with tracking, vehicle detection by parts using the L-SVM is presented in [17].

Detecting side-profile vehicles using a combination of parts has been explored in [18]. Using a camera looking out the side passenger's window, vehicles in adjacent lanes are detected by first detecting the front wheel, and then the rear. The combined parts are tracked using Kalman filtering.

For blind-spot detection of vehicles, a combination of parts approach has been explored in [19]. Using appearance and edge-based features, the parts form a vehicle using a Gaussian-weighted voting procedure to find the best configuration.

While prior works in vehicle detection by parts have shown promise, many require the 'root-filter' as part of detection, which can limit applicability to partial occlusions. Further there are few reported part-based vehicle detection systems that can run in real-time. Computational resources are dedicated to the feature extraction, part extraction, and evaluation of multiple configuration models, ie side vs. front. By contrast, the system presented in this work does not need to evaluate separate side and front models. Further, the system runs in real-time, with no specific hardware or software optimizations.

III. Detection of Vehicles Using Parts

We detect full vehicles by first detecting individual parts, and then matching parts that correspond to a given vehicle. During the part detection phase, parts are detected independently, using strong classifiers trained with active learning. During the matching phase, parts are matched to comprise full vehicles. The following subsections detail these operations. We defined the front part to comprise the portion of the vehicle including the bumper and front wheel. We defined the rear part to comprise the portion of the vehicle including the rear bumper and rear wheel.

A. Active Learning for Vehicle Part Detection

We train detectors for individual vehicle parts using an active learning framework [5]. An initial classifier is trained using labeled positive and negative examples. The initial classifier is evaluated on unlabeled on-road data. An interface allows the user to label detector responses as 'true positive' or 'false positive,' as well as adding any missed detections. These examples are archived for retraining, which builds an improved classifier [9].

We initialize the positive training set for front parts by using labeled faces of oncoming vehicles, and the front parts of side-profile vehicles. These training examples were hand-labeled. The goal is to develop one detector for the front part of a vehicle that is insensitive to the vehicle's orientation. Likewise, the training set for rear parts was initialized using the rear faces of preceding vehicles, and the rear parts of side-profile vehicles. The initial training set for each class consisted of 5000 positive and negative training examples. Active learning has been performed to improve recall-precision behavior for the front and rear part detection. We retrain with 5000 positive and negative training examples, and see improved detection and false positive rates.

Figure 2(a) shows two front part detection responses, resulting from evaluating the front detector on a given frame. The part on the left corresponds to the detected face of an oncoming vehicle. The part on the right corresponds to the front part of a side-profile vehicle. For part detection, we use

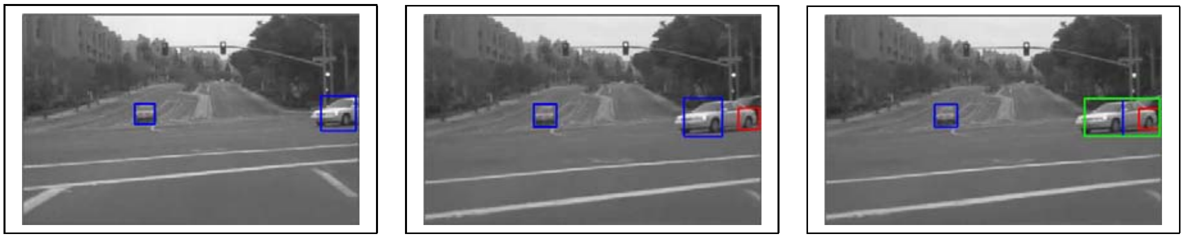

Fig. 2. Part-based vehicle detection, as a vehicle enters the camera's field of view. Detections of front vehicle parts are shown in blue. Detections of rear vehicle parts are shown in red. a) Two front parts are detected, corresponding to a fully-visible oncoming vehicle, and a partially occluded vehicle. b) Two front parts, and one rear part are detected. c) Using the matching classifier, one side-profile vehicle has been formed. The remaining unmatched front vehicle part is retained as an oncoming vehicle.

a boosted cascade of Haar-like features [20]. This classifier configuration evaluates quickly, and the feature set is well-suited for vehicle detection. Both front and rear parts have been trained with images patches of size 24×24 pixels.

B. Detecting Side Vehicles Using Parts

Given the part detection responses from evaluating the classifiers on a frame, we associate individual part responses to form vehicles. A given part can either be matched to form a side-profile vehicle, or retained as either an oncoming or preceding vehicle. We parametrize a rectangle p corresponding to a detected vehicle part, either front or rear, by the i, j position of the rectangle's top-left corner in the image plane, its width, and its height, as shown in equation 1.

$$p = \begin{bmatrix} i & j & w & h \end{bmatrix}^T \tag{1}$$

To match a pair of given parts, front to rear, we need to compute pertinent features. In this study, we compare the relative contributions of two sets of image features: geometric x_{geo}, and appearance-based x_{app}. The feature set x_{geo}, shown in equation 2 is a feature vector that encodes the relative displacements between a front part, p_f, and rear part, p_r, in the image plane, as well as their relative sizes. The parameter σ normalizes the distances to reference frame, which we have chosen to be scaled to the 24×24 size of image patches used in the training set.

$$\begin{aligned} x_{geo}(p_f, p_r) &= \begin{bmatrix} \frac{|i_f - i_r|}{\sigma} & \frac{|j_f - j_r|}{\sigma} & \frac{h_f}{h_r} \end{bmatrix}^T \\ \sigma &= \frac{w_f}{24.0} \end{aligned} \tag{2}$$

Geometric features can encode the spatial correspondence between front and rear parts of vehicles. The features in equation 2 are quick to compute. Using the absolute value of horizontal and vertical distances in the image plane allows the computed features to serve for left-facing or right-facing side-profile vehicles.

Appearance features can match corresponding parts of the same vehicle, using the similarity in their appearance. While there are myriad vision approaches to computing the similarity between image patches, we opted for a low-dimensional feature vector, that could be computed in near-real-time. We compute the appearance features x_{app} using a front part, p_f, and rear part, p_r. We compute the absolute difference of the mean intensities over the bounding boxes, the absolute difference of the median intensities over the bounding boxes, and the euclidean distance between their Histogram of Oriented Gradients feature vector [21]. For the HoG features, we use 8 orientations, 4 horizontal and 4 vertical cell assignments.

$$x_{app}(p_f, p_r) = \begin{bmatrix} |mean(p_f) - mean(p_r)| \\ |median(p_f) - median(p_r)| \\ ||HoG(front) - HoG(rear)||^2 \end{bmatrix} \tag{3}$$

$$\begin{aligned} f(x) &= \sum_i \alpha_i K(x, x_i) + b \\ y &= sgn(f(x)) \end{aligned} \tag{4}$$

We classify the part matching features using Support Vector Machine classification [6], as detailed in equation 4, where $K(.,.)$ is a kernel function. In the interest of speed, we attempted to use a linear kernel for both feature sets. As verified by cross-validation, the raw appearance features were not linearly separable, so we have used a radial basis kernel, as detailed in equation 5.

$$K(x, x_i) = e^{-\gamma ||x - x_i||^2} \tag{5}$$

We trained the geometric features SVM classifier using the linear kernel. This has the advantage that we can evaluate the SVM classifier using the primal form. As shown in equation 6, the primal form of the linear SVM classifier reduces to an inner product between the classifier's weight vector, and the feature vector. This is extremely useful for real-time implementation, as this inner product can be computed very quickly.

$$\begin{aligned} f(x) &= w^T x + b \\ y &= sgn(f(x)) \end{aligned} \tag{6}$$

The SVM classifiers were trained using the open-source LIBSVM software package [22]. 700 positive examples and

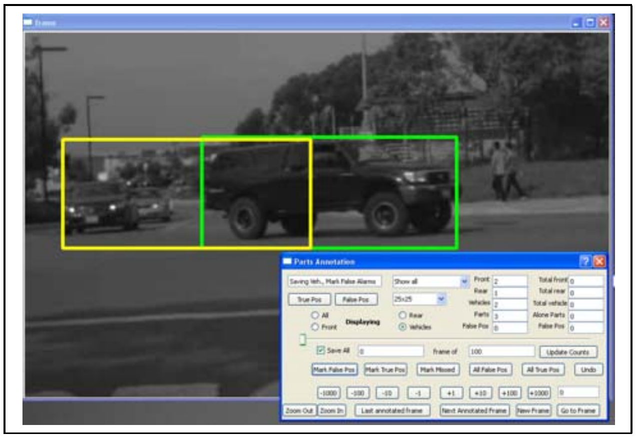

Fig. 3. The interface, querying which combinations of parts correspond to vehicles. A true vehicle [green], used as a positive training example, and a false vehicle [yellow], used as a negative training example.

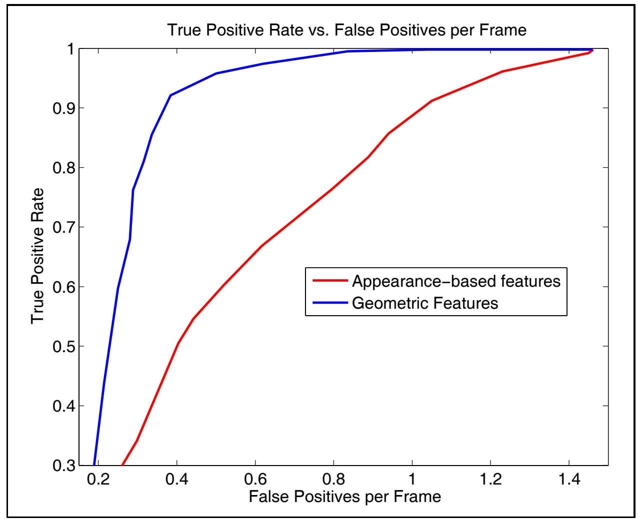

Fig. 4. True Positive Rate vs. False Positives per Frame. Appearance-based features [blue], and Geometric features [red].

700 negative examples were used for training each classifier. During active learning, the interface asked the user to select which part combinations comprised a side-profile vehicle, as shown in figure 3. Features were computed for these combinations, and archived as positive training examples. All other part combinations were archived as negative training examples.

IV. Experimental Evaluation

We evaluate the performance of the part-based vehicle detection classifiers over an evaluation set, captured on San Diego urban roads. The evaluation set consists of 500 consecutive frames, featuring 546 vehicles composed of parts, as well as 682 rear parts, and 715 front parts. The video was captured as the ego-vehicle approaches an intersection, and as it remains paused at a traffic signal. The video features oncoming, preceding, and cross traffic, as well as several partially occluded vehicles. The resolution of the frames is 500×312. Table I lists the color-coding for vehicle detections featured throughout this paper.

TABLE I
Color-Coding Used in Vehicle Detections

Blue	Front part of vehicles, Oncoming vehicles
Red	Rear part of vehicles, Preceding vehicles
Green	Full side-facing vehicle, as combination of parts

At the detection level, we do not distinguish between the front part of a vehicle and an oncoming vehicle. For an oncoming vehicle, we simply detect the front part, while its rear part remains occluded. Likewise, at the detection level, we do not distinguish between the rear part of a vehicle and a preceding vehicle. The direction in which a detected vehicle travels can be resolved with tracking [23].

Figure 4 plots the True Positive Rate versus the False Positives per Frame. We generate the performance curves by varying the decision threshold using equation 7; the parameters A and B are trained using maximum likelihood [24], [22].

$$p(y=1|x) = \frac{1}{1+e^{Af(x)+B}} \tag{7}$$

We plot the performance of the appearance-based features in red, and the geometric features in blue. As shown in the plot, the geometric features perform quite a bit better than the appearance-based features over the experimental test set. We have chosen appearance-based features with computation efficiency in mind, and attempted to use the image patch similarities as matching features. Given that the rear and front parts of a vehicle are not identical, and these differences will vary by the vehicle's design, it is likely that a more sophisticated, model-based appearance mapping must be used to properly match the rear and front parts of a vehicle.

In Table II we show the true positive rate, and false positives per frame for the part detectors. We note that the performance yields somewhat worse false positives per frame than [5]. We attribute this to the increased difficulty of detecting vehicle parts, versus fully-visible vehicles.

TABLE II
Part Detection Performance

Detector	True Positive Rate	False Positives per Frame
Front Part	88.5%	0.65
Rear Part	90.2%	0.52

We note that it was attempted to form a feature vector that combined appearance and geometric features, by simply concatenating the feature vectors, and training an SVM classifier. This model had extremely low detection rates. The authors believe that a better approach would involved decision fusion of the respective classifier responses [25]. As the geometric features yield better detection rates, false positive rates, and faster implementation, pursuit of system implementation using geometric features would likely be most fruitful.

TABLE III
PERFORMANCE COMPARISON, GEOMETRIC VS. APPEARANCE FEATURES

Feature Set	Kernel	Frames per Second
Appearance	RBF	5.8
Geometric	Linear	**14.7**

The geometric features, however, are simple to compute, and work fairly robustly, as shown by the blue curve in figure 4. Table III provides a comparison of kernel and computational cost associated with each part-based vehicle detector. The frames per second listed in the rightmost column includes the time spent evaluating the individual part detectors. We note that there are no specific hardware or software optimizations or parallelizations undertaken. For example, detection of independent parts could easily be parallelized, with associated increase in frames per second, but this sort of operation does not enter into table III.

We note that part-based vehicle detection using the geometric feature set is much computationally faster than the appearance feature set. There are two main reasons for this. Firstly, the geometric classifier features a linear kernel, the use of the primal form of the linear kernel reduces classification to a single inner product, which is very computationally efficient. Secondly, the appearance-based features include Histograms of Oriented gradients, which can be quite slow to compute. Indeed, [17] cites the use of HoG features as one of the major bottlenecks in their implementation.

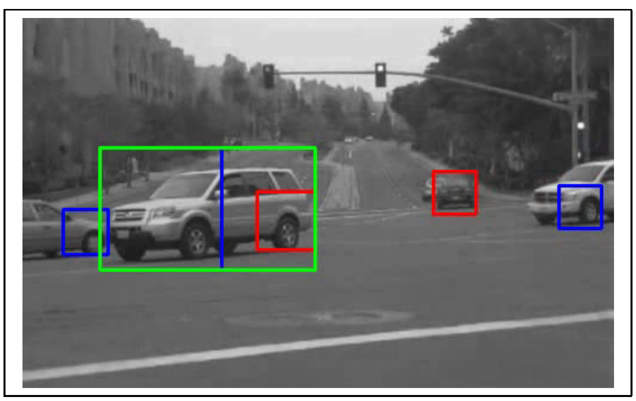

Fig. 5. Detecting partially occluded vehicles. The frame features fully-visible side profile vehicle [green], and three unmatched vehicle parts. Two of these correspond to vehicles entering the sensor's field of view, while the last one corresponds to a preceding vehicle.

Figure 5 shows a sample frame, featuring detection of partially-occluded vehicles using geometric features. There is one side-view vehicle that is fully unoccluded, which is detected and shown in green. There are two front vehicle parts detected in the frame, corresponding to vehicles that are entering the camera's field of view. There is one rear vehicle part, belonging to a preceding vehicle. A key benefit of the independent part detectors that we have trained, is the same detector can serve to detect the rear part of a side-facing or preceding vehicle. Likewise, the front part detector can detect the front face of an oncoming vehicle, or the front part of a side-facing vehicle.

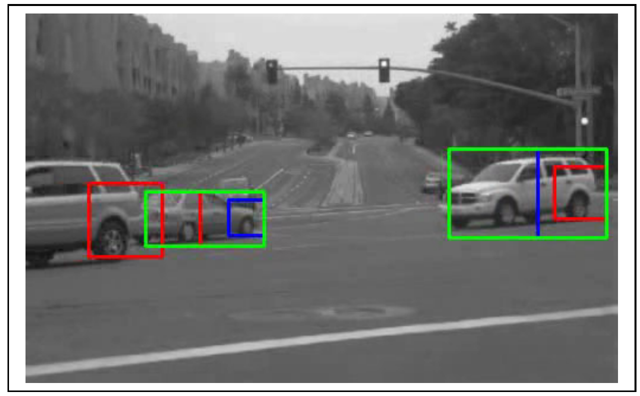

Fig. 6. Detecting partially occluded vehicles. The frame features two fully-visible side-facing vehicles, shown in green, and one rear part, belonging to a partially occluded vehicle, which is leaving the camera's field of view.

Figure 6 features two fully visible side-facing vehicles, detected using the geometric features, shown in green. The frame also features a detected rear part, corresponding to a vehicle that is leaving the camera's field of view, shown in red. Detection of such vehicles does not require a root filter, or for the vehicle to be fully visible. This is due to the independent evaluation of strong part classifiers.

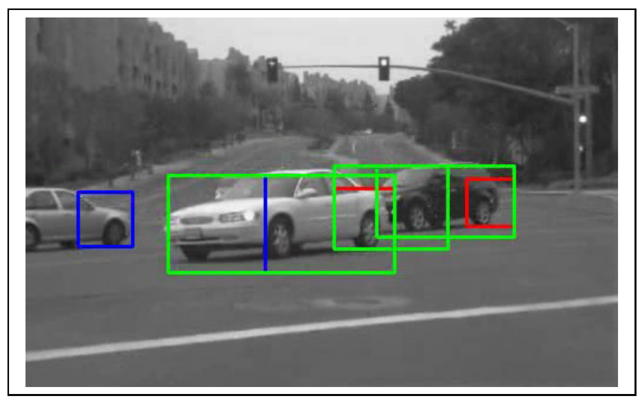

Fig. 7. False positive vehicle. A frame featuring two correctly detected vehicles, and one false positive.

Figure 7 shows a frame in which two side-facing vehicles have been correctly detected, but a false positive also resulted from evaluation of equation 6. In this case, parts that have been matched to form correct vehicles, have also been incorrectly matched across two vehicles. It is possible that by using equation 7, we can assign each part to the vehicle that yields the highest value of $p(y|x)$. The authors also envision an active learning approach to refining this problem, in which the user interactively provides feedback to the system. It is also possible that, given a tracking implementation, motion features could augment the classification accuracy of the system.

Figure 8 shows a frame in which a side-facing vehicle's parts have been detected, but evaluation of equation 6 missed the vehicle. The missed detection can likely be attributed to

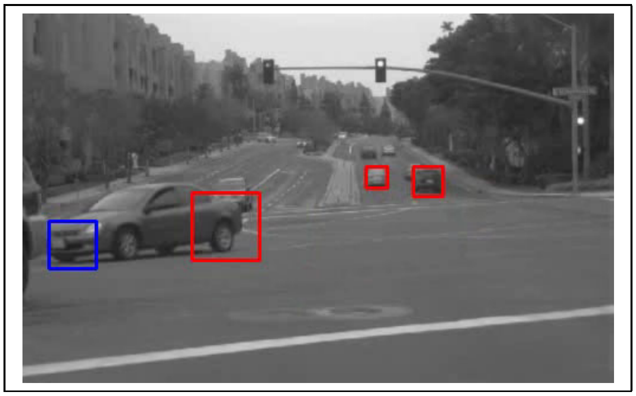

Fig. 8. While parts were correctly detected in this frame, the matching missed the side-view vehicle detection. There are two preceding vehicles detected, and one side-view vehicle missed.

the poor localization of the front part. The authors believe that implementing tracking for the individual parts can aid in such a situation, and that motion features could augment the geometric features used in this work. Further, the authors envision an active learning approach to help learn and refine the matching of parts based on user interactivity. Using a calibrated stereo rig, the part detection and tracking, as well as the vehicle detection and tracking, could be extended to stereo-vision, providing 3D spatial localization [23].

V. Concluding Remarks and Future Work

In this study, we have presented a novel and lightweight approach to on-road vehicle detection by parts. The focus of this research is geared towards the development of real-time systems that can detect oncoming, preceding, and side-profile vehicles, as well as handling partial occlusions. In contrast to other part-based vehicle detection works in the field, the proposed system runs in real time, with no specific hardware or software optimizations, and handles partial occlusions by design. Future work will include implementation of tracking, and the implications of motion features for part matching. An active learning formulation will be pursued, to better learn the matching function by user interactivity. Further, we will pursue implementation of vehicle tracking, extensions of part-based vehicle detection to stereo-vision [23], and learning of vehicle motion patterns in urban settings [26].

VI. Acknowledgments

The authors would like to thank colleagues Dr. Cuong Tran, Mr. Ashish Tawari, Mr. Larry Ly. This work was supported in part by Audi and Volkswagen Electronics Research Lab, Palo Alto, CA.

References

[1] D. of Transportation National Highway Traffic Safety Administration, "Traffic safety facts," 2011. [Online]. Available: http://www-nrd.nhtsa.dot.gov

[2] M. M. Trivedi and S. Y. Cheng, "Holistic sensing and active displays for intelligent driver support systems," *IEEE Computer*, 2007.

[3] T. Gandhi and M. M. Trivedi, "Parametric ego-motion estimation for vehicle surround analysis using an omnidirectional camera," *Machine Vision and Applications*, vol. 16, no. 2, pp. 85–95, feb 2005.

[4] S. Cheng and M. Trivedi, "Turn-intent analysis using body pose for intelligent driver assistance," *Pervasive Computing, IEEE*, vol. 5, no. 4, pp. 28 –37, oct.-dec. 2006.

[5] S. Sivaraman and M. M. Trivedi, "A general active learning framework for on-road vehicle recognition and tracking," *IEEE Trans. Intell Transp. Syst.*, 2010.

[6] C. Cortes and V. Vapnik, "Support vector networks," *Machine Learning*, 1995.

[7] A. Jazayeri, H. Cai, J. Y. Zheng, and M. Tuceryan, "Vehicle detection and tracking in car video based on motion model," *Intelligent Transportation Systems, IEEE Transactions on*, 2011.

[8] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: a review," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2006.

[9] S. Sivaraman and M. M. Trivedi, "Active learning for on-road vehicle detection: A comparative study," *Machine Vision and Applications*, 2011.

[10] B. Wu and R. Nevatia, "Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses," *International Journal of Computer Vision*, 2009.

[11] C. Huang and R. Nevatia, "High performance object detection by collaborative learning of joint ranking of granules features," in *Computer Vision and Pattern Recognition*, 2010.

[12] Z. Lin, G. Hua, and L. Davis, "Multiple instance feature for robust part-based object detection," in *Computer Vision and Pattern Recognition*, 2009.

[13] D. Tosato, M. Farenzena, M. Cristani, and V. Murino, "Part-based human detection on riemannian manifolds," in *International Conference on Image Processing*, 2010.

[14] D. Tosato, M. Farenzena, M. Cistani, and V. Murino, "A re-evaluation of pedestrian detection on riemannian manifolds," in *International Conference on Pattern Recognition*, 2010.

[15] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, September 2010.

[16] H. Niknejad, S. Mita, D. McAllester, and T. Naito, "Vision-based vehicle detection for nighttime with discriminately trained mixture of weighted deformable part models," in *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, 2011.

[17] H. Tehrani Niknejad, A. Takeuchi, S. Mita, and D. McAllester, "On-road multivehicle tracking using deformable object model and particle filter with improved likelihood estimation," *Intelligent Transportation Systems, IEEE Transactions on*, 2012.

[18] W.-C. Chang and C.-W. Cho, "Real-time side vehicle tracking using parts-based boosting," in *Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on*, 2008.

[19] B.-F. Lin, Y.-M. Chan, L.-C. Fu, P.-Y. Hsiao, L.-A. Chuang, S.-S. Huang, and M.-F. Lo, "Integrating appearance and edge features for sedan vehicle detection in the blind-spot area," *Intelligent Transportation Systems, IEEE Transactions on*, 2012.

[20] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *IEEE Computer Vision and Pattern Recognition Conference.*, 2001.

[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005.

[22] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[23] S. Sivaraman and M. M. Trivedi, "Combining monocular and stereo-vision for real-time vehicle ranging and tracking on multilane highways," *IEEE Intell. Transp. Syst. Conf.*, 2011.

[24] H.-T. Lin, C.-J. Lin, and R. C. Weng, "A note on platts probabilistic outputs for support vector machines," *Machine Learning*, 2007.

[25] S. Shivappa, M. M. Trivedi, and B. D. Rao, "Audio-visual information fusion in human computer interfaces and intelligent environments: A survey," *Proceedings of the IEEE*, 2010.

[26] S. Sivaraman, B. T. Morris, and M. M. Trivedi, "Learning multi-lane trajectories using vehicle-based vision," *International Conference on Computer Vision, Workshop on Computer Vision for Vehicle Technology*, 2011.