Vision-Based Behavior Prediction in Urban Traffic Environments by Scene Categorization

Martin Heracles[1][2]
heracles@cor-lab.uni-bielefeld.de
Fernando Martinelli[2]
fernando.martinelli@gmail.com
Jannik Fritsch[2]
jannik.fritsch@honda-ri.de

[1] CoR-Lab
Bielefeld University
Germany

[2] Honda Research Institute Europe
Offenbach/Main
Germany

Abstract

We propose a method for vision-based scene understanding in urban traffic environments that predicts the appropriate behavior of a human driver in a given visual scene. The method relies on a decomposition of the visual scene into its constituent objects by image segmentation and uses segmentation-based features that represent both their identity and spatial properties. We show how the behavior prediction can be naturally formulated as scene categorization problem and how ground truth behavior data for learning a classifier can be automatically generated from any monocular video sequence recorded from a moving vehicle, using structure from motion techniques. We evaluate our method both quantitatively and qualitatively on the recently proposed CamVid dataset, predicting the appropriate velocity and yaw rate of the car as well as their appropriate change for both day and dusk sequences. In particular, we investigate the impact of the underlying segmentation and the number of behavior classes on the quality of these predictions.

1 Introduction

Understanding complex visual scenes such as urban traffic environments is a challenging task that humans are able to solve with the greatest of ease. As pedestrians, car drivers etc., they convincingly demonstrate every day that this can be achieved largely on the basis of (stereo) vision, without requiring additional information from laser scanners, infrared sensors, GPS or map data of the environment. In this paper, we take the perspective of a human driver and address the problem of vision-based scene understanding from video sequences that are acquired by a car-mounted camera, monitoring the scene in front of the car while driving. In particular, we are interested in predicting the appropriate driving behavior from the current visual scene. Such capability is highly important for future driver assistance systems and autonomous driving applications since it provides the car with expectations of what a human driver would normally do in a given traffic situation. Depending on the application, these expectations could be compared to the actual behavior of the human driver, communicating a warning in case of significant deviation, or used directly to control the effectors of the car.

BMVC 2010 doi:10.5244/C.24.71

Background. Due to the enormous complexity of unconstrained real-world traffic environments, existing driver assistance systems restrict themselves to very specific tasks such as lane keeping, collision avoidance, or automatic following, processing only very limited subsets of information about the scene. Autonomous driving applications go beyond such individual tasks and have already demonstrated impressive levels of performance and robustness, the DARPA Challenges [6, 7] being two prominent examples. However, the scene understanding capabilities of such systems typically rely on extensive technical sensor setups involving laser scanners, infrared sensors, radar sensors, GPS and annotated map data of the environment, with vision playing only a subordinate role.

In contrast, we consider scene understanding based on visual information alone, without the use of additional sensors or knowledge about the environment, and formulate the behavior generation as scene categorization problem (Sec. 2). Early work on scene categorization considered the scene as a whole [29, 30] or at the level of coarse sub-blocks [24, 27] and enabled a basic distinction between a small number of well-distinguishable categories, based on global or block-wise image statistics, respectively. In analogy to text documents, bag-of-words approaches model images as histograms of local patch descriptors [19] and have been successfully applied to scene categorization tasks involving up to 13 classes [10, 21]. Their lack of representing spatial properties has been addressed by Lazebnik *et al.* [17] who proposed a spatial pyramid representation, consisting of a hierarchy of bag-of-words representations over increasingly fine-grained image grids. Approaches considering object-centered rather than grid-based spatial properties [20] overcome the implicit assumption of low variance in the spatial scene layouts but are computationally expensive. Drawing inspiration from text document analysis and texture models, latent topic models such as pLSA [14] and LDA [1] have been shown to improve the bag-of-words models by capturing co-occurrences between codewords [2, 10, 22] and have evolved into complex generative models capable of simultaneous scene categorization, segmentation and annotation [5, 18, 31]. In general, however, scene categorization methods have focused on rather generic, domain-independent categories, without direct application to traffic scene understanding or behavior prediction.

Discriminative methods for image segmentation, in contrast, have explicitly dealt with traffic scene understanding, conditional random fields [16] being among the most popular and successful [26, 28]. While traditional approaches have mainly focused on appearance-based image features such as texture [25] and edge orientation histograms [8], Brostow *et al.* [3] have recently shown that a reasonable segmentation can already be inferred from sparse 3D features derived from noisy 3D point clouds. State-of-the-art methods [26] combine both 3D and appearance-based features and also incorporate higher order potentials [15], which exploit an unsupervised initial segmentation for more accurate object boundaries. However, the segmentation problem has mostly been studied in isolation. Notable exceptions are Wang *et al.* [32] who consider segmentation and scene categorization in a common framework, and Ess *et al.* [9] who use segmentation as a basis for road layout categorization and object detection. We build on the work of the latter by employing the same set of features (Sec. 4), but we use them to actually predict the appropriate driving behavior rather than the upcoming road layout, which involves much more fine-grained categories.

Contributions. To summarize, our paper makes the following important contributions. *(i)* We extend the approach of Ess *et al.* [9] to actual behavior prediction, both in terms of the appropriate velocity and yaw rate of the car as well as their appropriate change (Sec. 2). *(ii)* We provide detailed predictions of these quantities for both day and dusk sequences of the recently proposed CamVid dataset [4], including an analysis of the impact of the underlying segmentation and the number of behavior classes on the prediction quality (Sec. 5).

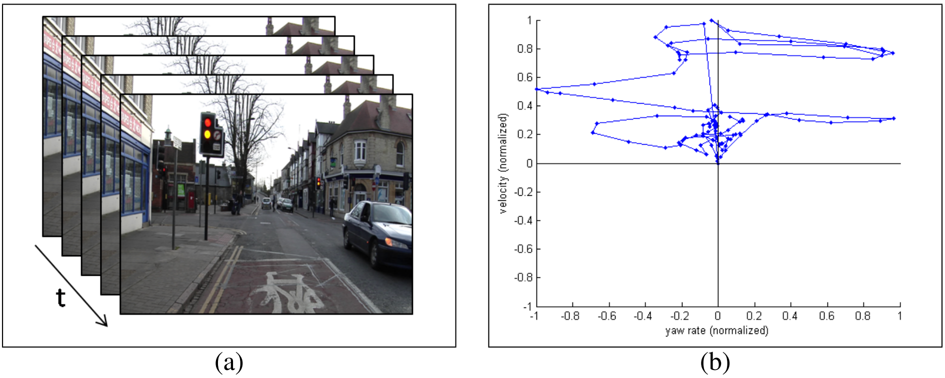
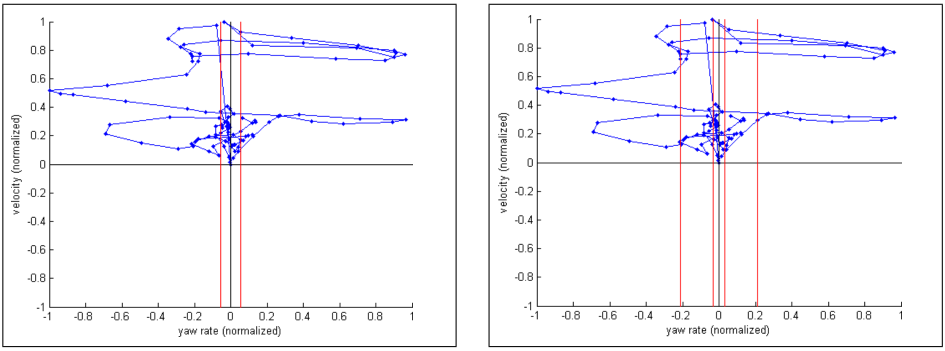


Figure 1: (a) Uncalibrated monocular image sequence, acquired by a car-mounted camera while driving. (b) Corresponding behavior of the driver in terms of velocity and yaw rate of the car, estimated using structure from motion techniques (Sec. 3). From these, our method learns to predict the appropriate velocity and yaw rate of the car (as well as their appropriate change) for previously unseen images. Note that (b) is automatically computed from (a), i.e., our method only requires an image sequence for learning the prediction.

2 Problem formulation

Given an uncalibrated monocular image sequence $I = (i_1, \dots, i_n)$ (Fig. 1a), it is well-known that both the ego-motion of the camera and the 3D structure of the scene can be reconstructed up to an unknown scale factor by structure from motion techniques [13]. While the 3D structure of the scene has been shown to be useful for image segmentation, among others [3], less attention has been paid to the ego-motion of the camera in this context. For image sequences acquired by a car-mounted camera while driving, however, the ego-motion of the camera largely corresponds to the motion of the car as a physical body, provided that the camera is firmly mounted. In particular, it is possible to infer the velocity $v \in [-1, 1]$ and yaw rate $y \in [-1, 1]$ of the car from the camera trajectory $C = (c_1, \dots, c_n)$ (Sec. 3). Note that this is subject to the unknown scale factor as well, hence the normalization.

As a result, the camera trajectory has a corresponding trajectory $B = ((v_1, y_1), \dots, (v_n, y_n))$ in the 2D space spanned by velocity and yaw rate (Fig. 1b). In particular, for each camera pose $c_k \in C$ we have the corresponding image $i_k \in I$ on the one hand and the velocity and yaw rate $(v_k, y_k) \in B$ of the car on the other hand. While i_k largely depicts what the driver currently sees, the pair (v_k, y_k) largely reflects what the driver currently does: Although we cannot directly observe how the driver interacts with the steering wheel, throttle and brake pedals, we know that this ultimately affects the velocity and yaw rate of the car, hence it is sufficient for our purposes to observe (infer) only these. Clearly, the behavior of the driver depends to a great extent on what he or she currently sees, i.e., the pair (v_k, y_k) has a visible correlate in i_k. Intuitive examples include braking because of a red traffic light or an obstacle, or steering left because of a curve. Our goal is to learn such correlations from I and B, which can be seen as examples of the appropriate behavior in different visual scenes provided by a responsible driver, in order to be able to predict the appropriate behavior $(v^*, y^*) \in [-1, 1]^2$ for previously unseen images $i^* \notin I$. Note that inferring the appropriate behavior by structure from motion techniques is only possible while learning, since only then an image sequence is available that may justly be assumed to be provided by a responsible driver.



(a) (b)

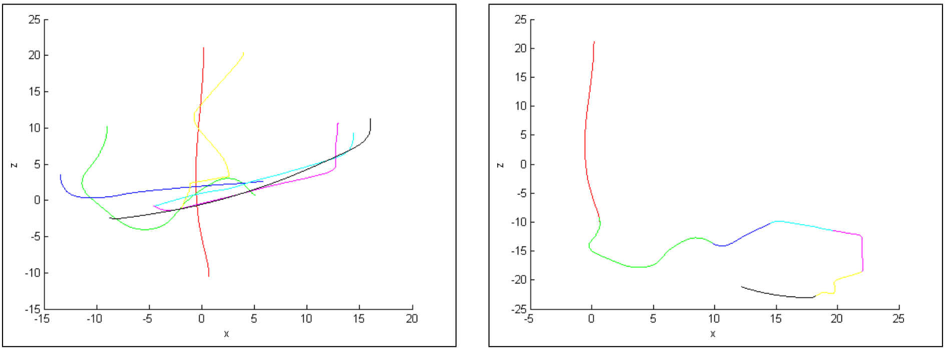
Figure 2: The behavior space is sub-divided into an arbitrary number of behavior categories by symmetric thresholds (red), thereby inducing visual categories on the corresponding images (Fig. 1a). Examples of behavior categories for yaw rate: (a) Basic distinction between "steering left", "driving straight", and "steering right". (b) More fine-grained distinction between "strongly left", "slightly left", "straight", "slightly right", and "strongly right".

Scene categorization. In order to make the behavior prediction approachable, we formulate it as scene categorization problem. This is achieved by imposing a limited number of well-defined thresholds on the behavior data B. More exactly, we consider velocity, yaw rate and their respective changes independently from each other, thus predicting a 1D quantity $q \in [-1,1]$ each, and define symmetric thresholds $-t_p,\dots,-t_1,t_1,\dots,t_p \in [-1,1]$ on the domain of q (Fig. 2), thus sub-dividing B into behavior classes $B_0 = \{(v,y) \in B | -t_1 < q < t_1\}$, $B_{t_1} = \{(v,y) \in B | t_1 \leq q < t_2\}$, $B_{-t_1} = \{(v,y) \in B | -t_2 > q \geq -t_1\}$ etc. We set the thresholds such that the behavior classes are balanced, i.e., $|B_{-t_p}| = \cdots = |B_{t_p}|$. Since each $(v_k,y_k) \in B$ has a corresponding image $i_k \in I$, these induce image classes $I_0 = \{i_k \in I | (v_k,y_k) \in B_0\}$, $I_{t_1} = \{i_k \in I | (v_k,y_k) \in B_{t_1}\}$, $I_{-t_1} = \{i_k \in I | (v_k,y_k) \in B_{-t_1}\}$ etc. By classifying a new image $i' \notin I$ as belonging to a certain I_* based on its similarity to the $i \in I_*$, the behavior represented by B_* is then taken to be the appropriate behavior for i'. In general, a high number of behavior classes is desirable since it allows for a more fine-grained distinction of the appropriate behavior in a given situation i', however, this also increases the difficulty of the problem because the classes become more similar. We investigate this dependency in Sec. 5.

3 Training data generation

The behavior data B corresponding to the image sequence I from which to learn the prediction is inferred automatically from I by using standard structure from motion techniques [3, 4, 13]). For completeness, we briefly summarize the main steps involved, following [3].

First, keypoint detection using the Harris corner detector [12] is performed on the individual images $i \in I$, yielding a sparse set of 2D points per image. Next, correspondence between keypoints of subsequent images is established by comparing local image patches centered at the keypoints via Normalized Cross Correlation, resulting in short trajectories of keypoints that correspond to the same 3D point in the scene. The coordinates of the 3D points and the camera pose are then inferred by triangulation, assuming that the 3D points are static. In the uncalibrated case, however, this 3D reconstruction is inherently ambiguous and can only be achieved up to an unknown scale factor.



(a) (b)

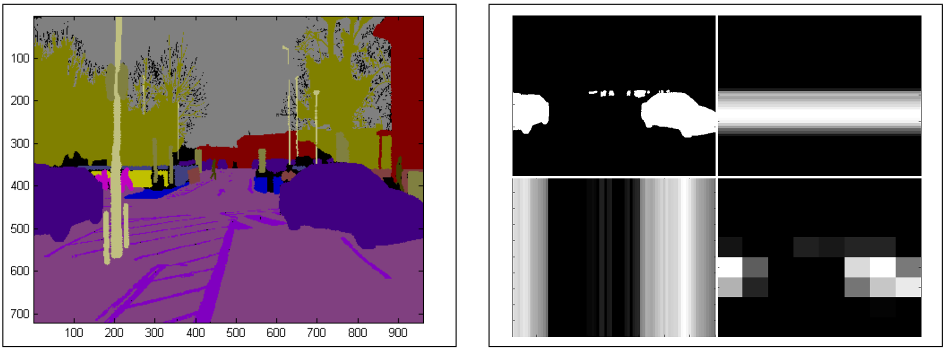
Figure 3: Most of the CamVid sequences [4] are split into independent batches. In order to establish a common scale and avoid artifacts in the inferred behavior data, we align the batches with each other by exploiting short overlaps. The alignment includes translation, rotation and scaling of the individual batches. (a) Before alignment. (b) After alignment. (Camera trajectories of different batches shown in different colors.)

Batch alignment. The global scale ambiguity implies that we can only infer the relative velocity and yaw rate of the car rather than their absolute values. In addition, the 3D reconstruction step often involves breaking down the sequence into independent batches (Fig. 3a), which is also true for the CamVid dataset [4]. Before inferring the behavior data, we therefore perform a batch alignment in order to establish a common scale across the batches. We exploit that the last $L = (c_{b_1}, \dots, c_{b_n})$ and the first $F = (c_{p_1}, \dots, c_{p_n})$ camera positions of two subsequent batches b and p are overlapping. First, we compute the translation vector $t = g_b - g_p$ that shifts F to L, where g_b and g_p denote their respective centers of gravity, and apply t to the entire batch p. Next, we compute direction vectors $d_b = g_{b_2} - g_{b_1}$ and $d_p = g_{p_2} - g_{p_1}$ of L and F, respectively, where g_{b_1}, g_{b_2} and g_{p_1}, g_{p_2} denote the centers of gravity of the first and last half of L and F, respectively. We apply the rotation that turns d_p into d_b to the entire batch p. Last, we compute the scale factor $s = \|d_b\| / \|d_p\|$ that aligns the spacing of the $c_p \in F$ to that of the $c_b \in L$ and apply it to p. The result is depicted in Fig. 3b.

The batch alignment serves to establish a common scale, which is sufficient for our purposes, not to estimate the global scale of the scene. In practice, it works sufficiently reliable, however, it implicitly requires that the overlap does not coincide with the car standing. Since the third day sequence of the CamVid dataset (VD sequence) is affected by this issue, and also because of the page limit, we chose to focus on the other three CamVid sequences, including the dusk sequence. It is worth noting that a method for estimating the global scale by exploiting non-holonomicity constraints has been recently proposed [23].

4 Segmentation-based features

Having inferred the behavior data B corresponding to the image sequence I from which to learn the prediction, the actual learning uses segmentation-based image features, representing both identity and spatial properties of the objects in the scene. We use the same set of features as recently proposed by Ess *et al.* [9], since these have already been shown to be suitable for coarse road layout categorization, and investigate their suitability for actually predicting the appropriate driving behavior, which involves more fine-grained classes. We



(a) (b)

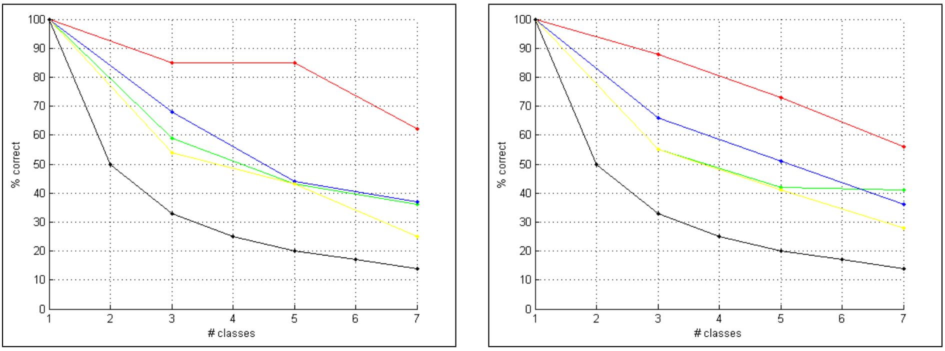
Figure 4: Behavior prediction is learned using features derived from a segmentation of the image. (a) Example segmentation (CamVid ground truth), and (b) example showing the row, column and 8x8 spatial layout features for cars. Such features are computed for each of the 12 object classes considered and form a large feature vector that represents the image.

did not include the periodicity features since these mainly serve to detect repetitive objects like zebra crossings, for example, which is not our focus. For completeness, we briefly summarize the features we use, following [9].

Given a segmentation s of an image $i \in I$ that assigns an object class label $s(u,v) \in \{l_1,\ldots,l_n\}$ to each image pixel (Fig. 4a), we first compute n binary maps $s_{l_1},\ldots,s_{l_n}$ such that $s_{l_*}(u,v) = 1$ iff $s(u,v) = l_*$ (Fig. 4b, top left). Each of the s_{l_*} is then downsampled into an 8x8, 4x4, and 2x2 feature map $\overline{s_{l_*}}$ by subdividing s_{l_*} into the corresponding number of blocks and computing the average pixel value for each block (Fig. 4b, bottom right). Intuitively, the $\overline{s_{l_*}}$ represent the coarse spatial layout of objects in the scene. Similarly, we downsample each of the s_{l_*} into an 80x60 map from which we then compute a row feature vector $\overline{r_{l_*}}$ and a column feature vector $\overline{c_{l_*}}$ by averaging over its rows and columns, respectively (Fig. 4b, top right and bottom left). Intuitively, the $\overline{c_{l_*}}$ are related to the yaw rate (avoiding obstacles) and the $\overline{r_{l_*}}$ are related to the velocity (approaching objects). Finally, we combine the two binary maps corresponding to lane markings and curbs by computing their pixel-wise maximum, where curbs are defined as boundaries between road and sidewalk in s, and apply a gradient operator (Sobel filter) to the combined map. The resulting gradient image is then subdivided into 4x4 blocks and we compute an 18-bin edge orientation histogram per block. In the end, the $\overline{s_{l_*}}$, $\overline{r_{l_*}}$, $\overline{c_{l_*}}$ and histograms are all serialized into one large feature vector of size $n(64+16+4)+n(60+80)+16\cdot 18$ that represents the image i.

5 Results

We evaluate our method on the recently proposed CamVid dataset [4]. The CamVid dataset consists of four video sequences that have been recorded by a car-mounted camera while driving naturally in crowded urban traffic environments. The color images thus obtained are available at a framerate of 30 Hz and have a resolution of 960x720 pixels. In addition, each image is accompanied by a pre-computed set of 2D keypoints obtained by Harris corner detection [12] as well as the corresponding 3D point cloud and camera pose obtained by structure-from-motion techniques. For every 30th image, a high-quality ground truth segmentation is available that distinguishes between 32 object classes at pixel-level (Fig. 4a).



(a) (b)

Figure 5: Behavior prediction results on TP sequence, using (a) CamVid segmentation and (b) CRF segmentation. Baseline (black) corresponds to random guessing. (red: velocity, green: yaw rate, blue: velocity change, yellow: yaw rate change)

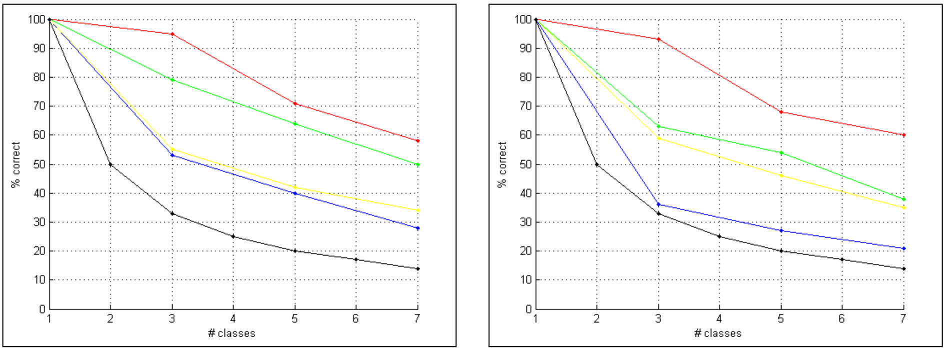
Predicted quantity	% correct (3 classes)		% correct (5 classes)		% correct (7 classes)	
Velocity	85 %	88 %	85 %	73 %	62 %	56 %
Yaw rate	59 %	55 %	43 %	42 %	36 %	41 %
Velocity change	68 %	66 %	44 %	51 %	37 %	36 %
Yaw rate change	54 %	55 %	43 %	41 %	25 %	28 %
Baseline	33 %		20 %		14 %	

Table 1: Behavior prediction results on TP sequence, corresponding to the graphs in Fig. 5. All values obtained by 10-fold cross-validation. Performance of best classifiers shown.

Our evaluation considers 12 of these classes (buildings, cars, poles, lane markings, pedestrians, road, sidewalks, signs, sky, trees, vegetation, and walls), matching those of [9].

The evaluation consists of two steps. In the first step, we use the CamVid ground truth segmentation as a basis for our behavior prediction. This enables us to decouple the influence of the features on the quality of the predictions from the influence of the underlying segmentation, thus providing insight into the quality of the features. At the same time, it establishes an upper bound on the prediction quality, assuming an ideal segmentation. In the second step, we relax this assumption and replace the CamVid ground truth segmentation by a realistic segmentation, using a Conditional Random Field with unary and pairwise potentials, based on color, edge, location and texton features [25]. Its overall accuracy on the 12 classes is about 65 %, which is lower than a state-of-the-art segmentation [26] due to the lack of texton spatial layout and higher order potentials, but is trained in the same way. For a lower bound on the prediction quality, uninformed random guessing serves as a baseline.

We consider each of the CamVid sequences separately (without the VD sequence, for the reasons given at the end of Sec. 3) and predict the velocity, yaw rate, velocity change and yaw rate change. To do so, we first compute ground truth behavior data for each of these quantities, using the camera poses provided with the respective sequence as described in Sec. 3. Since the CamVid ground truth segmentation is available for every 30th image only, we compute the behavior ground truth data from the camera poses coresponding to these images, although the camera poses are available for every image. This avoids artifacts arising from different sampling rates and, in practice, results in sufficiently accurate behavior trajec-



(a) (b)

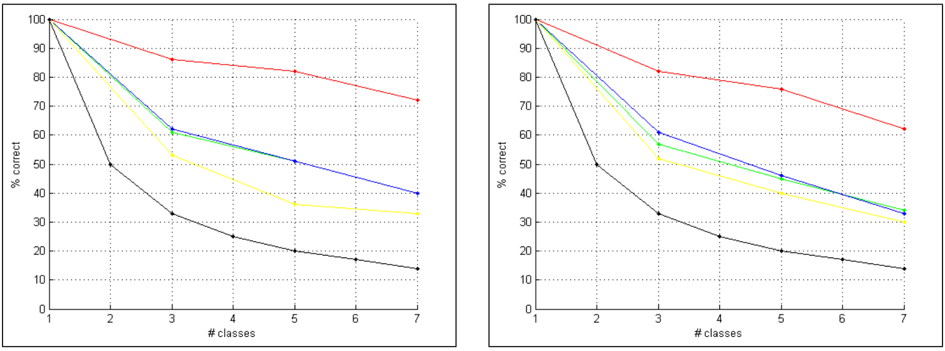
Figure 6: Behavior prediction results on R0 sequence, using (a) CamVid segmentation and (b) CRF segmentation. Baseline (black) corresponds to random guessing.

Predicted quantity	% correct (3 classes)		% correct (5 classes)		% correct (7 classes)	
Velocity	95 %	93 %	71 %	68 %	58 %	56 %
Yaw rate	79 %	63 %	64 %	54 %	50 %	38 %
Velocity change	53 %	36 %	40 %	27 %	28 %	19 %
Yaw rate change	55 %	59 %	42 %	46 %	34 %	35 %
Baseline	33 %		20 %		14 %	

Table 2: Behavior prediction results on R0 sequence, corresponding to the graphs in Fig. 6. All values obtained by 10-fold cross-validation. Performance of best classifiers shown.

tories. Having computed the behavior ground truth data for each of the four quantities to be predicted, we compute the segmentation-based feature vectors from the segmentation data corresponding to these images as described in Sec. 4. Depending on whether we are in the first or second part of the evaluation, we use either the CamVid ground truth segmentation or the Conditional Random Field segmentation in this step. We then define a 3-, 5-, and 7-class image categorization problem for each of the quantities to be predicted by quantizing the respective continuous values of the behavior ground truth data into the corresponding number of classes, as described in Sec. 2. For each of these categorization problems, we choose the thresholds such that all classes are approximately equally probable, which ensures that the baseline mentioned above is applicable. Finally, we learn a 3-, 5-, and 7-class classifier built from one-versus-all classifiers. We use GentleBoost [] with decision stumps as weak classifiers. For each (multi-)classifier, the optimal number of weak classifiers is determined by 10-fold cross-validation on the entire sequence, and the resulting classification performances are reported: Table 1 and Fig. 5 show the results for the TP sequence (dusk), Table 2 and Fig. 6 show the results for the R0 sequence (day), and Table 3 and Fig. 7 show the results for the E5 sequence (day).

Discussion. As expected, the prediction quality on all sequences decreases as the number of behavior classes increases since the classes become more difficult to distinguish. In general, the prediction quality for all quantities is consistently much higher than the baseline. However, even when using the CamVid segmentation, the prediction quality remains much lower than the theoretical optimum of 100%. One reason is that not all behavior of the driver may have a visible correlate in the visual scene, but it also suggests that the represen-



(a) (b)

Figure 7: Behavior prediction results on E5 sequence, using (a) CamVid segmentation and (b) CRF segmentation. Baseline (black) corresponds to random guessing.

Predicted quantity	% correct (3 classes)		% correct (5 classes)		% correct (7 classes)	
Velocity	86 %	83 %	82 %	76 %	72 %	63 %
Yaw rate	61 %	57 %	51 %	45 %	40 %	35 %
Velocity change	62 %	61 %	51 %	46 %	40 %	36 %
Yaw rate change	53 %	52 %	36 %	40 %	33 %	30 %
Baseline	33 %		20 %		14 %	

Table 3: Behavior prediction results on E5 sequence, corresponding to the graphs in Fig. 7. All values obtained by 10-fold cross-validation. Performance of best classifiers shown.

tational power of the features is not yet sufficient. When using the CamVid segmentation, the results are similar on all sequences. An exception is the yaw rate prediction on the R0 sequence (Fig. 6a), which is better than for the other sequences. We attribute this to the large number of curves in the R0 sequence, providing more training data for the yaw rate. When using the CRF segmentation, the results are also quite similar on all sequences, except for the velocity change prediction on the R0 sequence (Fig. 6b), which almost drops to baseline level. We attribute this to the fact that the R0 sequence is very different from the E5 and VD sequence used for training, being less road-like and more crowded, and conclude that it was hard for the velocity change prediction to generalize to the R0 sequence. Note that some values are higher than for the CamVid segmentation, which should not be the case. We attribute this to the variance when randomly choosing training and test sets despite the 10-fold cross-validation and expect this effect to become less pronounced with a higher number of folds. Overall, we observe that the difference between using the CamVid segmentation and the realistic segmentation is much lower than expected.

6 Conclusion

Our experiments show that the segmentation-based features proposed by Ess *et al*. [9] are not only suitable for scene classification with respect to rather coarse-grained categories such as the upcoming road layout but also can be used to actually predict the appropriate driving behavior in terms of the velocity and yaw rate of the car and their appropriate change. Further, by comparing the predictions obtained when using an ideal segmentation to those obtained when using a realistic segmentation, we were able to show that the prediction quality does not so much depend on the quality of the underlying segmentation. We conclude that more expressive features need to be sought rather than a more accurate segmentation as far as behavior prediction is concerned.

Consequently, future work should concentrate on the feature space in order to improve the performance, taking into account additional features such as 3D information and optical flow, for example. Also, a more extensive evaluation on longer video sequences should be conducted in order to estimate the performance in real-world applications.

References

[1] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] A. Bosch, A. Zisserman, and X. Munoz. Scene classification via pLSA. In *Proc. ECCV*, 2006.

[3] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *Proc. ECCV*, 2008.

[4] G. J. Brostow, J. Fauqueur, and R. Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2):88–97, 2009.

[5] L. Cao and L. Fei-Fei. Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. In *Proc. ICCV*, 2007.

[6] DARPA Grand Challenge, 2005. URL `www.darpa.mil/grandchallenge05`.

[7] DARPA Urban Challenge, 2007. URL `www.darpa.mil/grandchallenge`.

[8] N. Dalal and B. Triggs. Histogram of oriented gradients for human detection. In *Proc. CVPR*, 2005.

[9] A. Ess, T. Muller, H. Grabner, and L. van Gool. Segmentation-based urban traffic scene understanding. In *Proc. BMVC*, 2009.

[10] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proc. CVPR*, 2005.

[11] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: A statistical view of boosting. *Annals of Statistics*, 28:337–407, 2000.

[12] C. Harris and M. Stephens. A combined corner and edge detector. In *ALVEY Vision Conference*, 1988.

[13] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. 2003.

[14] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 41(2):177–196, 2001.

[15] P. Kohli, L. Ladicky, and P. H. S. Torr. Robust higher order potentials for enforcing label consistency. *IJCV*, 82:302–324, 2009.

[16] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. ICML*, 2001.

[17] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proc. CVPR*, 2006.

[18] L. J. Li, R. Socher, and L. Fei-Fei. Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. In *Proc. CVPR*, 2009.

[19] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[20] B. Ommer and J. M. Buhmann. Learning the compositional nature of visual objects. In *Proc. CVPR*, 2007.

[21] F. Perronin, C. Dance, G. Csurka, and M. Bressan. Adapted vocabularies for generic visual categorization. In *Proc. ECCV*, 2006.

[22] P. Quelhas, F. Monay, J. M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. van Gool. Modeling scenes with local descriptors and latent aspects. In *Proc. ICCV*, 2005.

[23] D. Scaramuzza, F. Fraundorfer, M. Pollefeys, and R. Siegwart. Absolute scale in structure from motion from a single vehicle mounted camera by exploiting nonholonomic constraints. In *Proc. ICCV*, 2009.

[24] N. Serrano, A. Savakis, and J. Luo. Improved scene classification using efficient low-level features and semantic cues. *Pattern Recognition*, 37:1773–1784, 2004.

[25] J. Shotton, J. M. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *Proc. ECCV*, 2006.

[26] P. Sturgess, K. Alahari, L. Ladicky, and P. H. S. Torr. Combining appearance and structure from motion features for road scene understanding. In *Proc. BMVC*, 2009.

[27] M. Szummer and R. W. Picard. Indoor-outdoor image classification. In *ICCV Workshop on Content-Based Access of Image and Video Databases*, 1998.

[28] M. Turtinen and M. Pietikainen. Contextual analysis of textured scene images. In *Proc. BMVC*, 2006.

[29] A. Vailaya, A. Jain, and H. Zhang. On image classification: City vs. landscapes. *Pattern Recognition*, 31(12):1921–1935, 1998.

[30] A. Vailaya, A. Figueiredo, A. Jain, and H. Zhang. Image classification for content-based indexing. *IEEE Transactions on Image Processing*, 10:117–129, 2001.

[31] C. Wang, D. Blei, and L. Fei-Fei. Simultaneous image classification and annotation. In *Proc. CVPR*, 2009.

[32] Y. Wang and S. Gong. Conditional random field for natural scene categorization. In *Proc. BMVC*, 2008.