Proceedings of the 2011 IEEE
International Conference on Robotics and Biomimetics
December 7-11, 2011, Phuket, Thailand

Stereo-based Road Boundary Tracking for Mobile Robot Navigation

Takeshi Chiku Jun Miura Junji Satake
Department of Computer Science and Engineering
Toyohashi University of Technology, Toyohashi, Japan

***Abstract*— This paper describes a method of stereo-based road boundary tracking for mobile robot navigation. Since sensory evidence for road boundaries might change from place to place, we cannot depend on a single cue but have to use multiple sensory features. The method uses color, edge, and height information obtained from a single stereo camera. To cope with a variety of road types and shapes and that of their changes, we adopt a particle filter in which road boundary hypotheses are represented by particles. The proposed method has been tested in various road scenes and conditions, and verified to be effective for autonomous driving of a mobile robot.**

I. Introduction

Outdoor navigation has been one of the fundamental research areas in robotics. To realize a fully autonomous navigation, a robot has to have many functions such as route planning, localization, road detection and following, and obstacle avoidance. This paper focuses on road detection.

Various approaches have been proposed for detecting roads or traversable regions. Examples are: detecting road boundaries using vision [1], [2], extracting road regions using color [3], [4], road plane extraction using stereo [5], detecting road boundary features using range sensors [6], [7], and detecting traversable regions using multiple sensors [8].

In a well-prepared road scene such as an expressway, a specific feature (e.g. white lane markers) can be used for road detection. Since effective sensory information varies from place to place, however, it is indispensable to use multiple sensory features. It is also necessary to temporarily integrate sensor data to cope with occasional sensing failures and missing effective features using, for example, Kalman filters [2], particle filters [9], [10], or swarm-based region detection [11].

We have developed a robust road boundary modeling method using a camera and a 2D laser range finder [12]. In this method, multiple sensory features (color, edge, and range) and flexible road models are effectively integrated in the particle filter framework. We also performed preliminary experiments using a stereo camera, instead of the laser range finder, for obtaining shape information of road boundaries. In this paper, we extend the method with an improved road model for stereo data and extensively evaluate its performance by road detection and navigation experiments in various scenes.

The rest of the paper is organized as follows. Sec. II explains the boundary tracking method we have developed and its extension. Sec. III describes results of road boundary modeling experiments with the one to show the effectiveness of estimating the gaps between image features. Sec. IV shows navigation experiments for various road scenes. Sec. V concludes the paper and discusses future work.

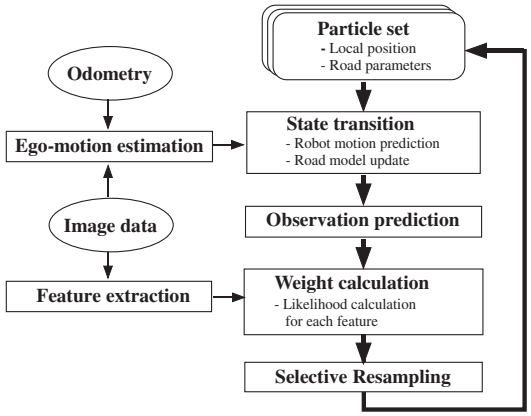


Fig. 1. Overview of the proposed method.

II. Road boundary tracking method

A. Overview of the method

The proposed method adopts a particle filter [13] for integrating multiple sensory information and for managing the road shape and type changes. Fig. 1 shows an overview of the method. The right-hand side of the figure indicates the iteration of particle filter, while the left-hand side indicates the sensor data processing. We use color, edge, and height features obtained from a stereo camera.

B. Road model and state vector

1) Unbranched and branching models: A state vector has both the robot position and the road parameters, with respect to the previous robot pose, for their simultaneous estimation. The robot position is equivalent to the ego-motion from the previous position, which is represented by 2D translation and the rotation.

We use two types of models: unbranched model and branching models (see Figs. 2 and 3). A model basically consists of a set of *road segments*, each of which is either linear or circular type. A branching model additionally has a branching part.

2) Representing gaps between image feature positions: Fig. 4 shows a road scene where different boundaries are supported by different features. Such gaps often exist but but they can be considered constant in a local region. We

978-1-4577-2137-3/11/$26.00 © 2011 IEEE

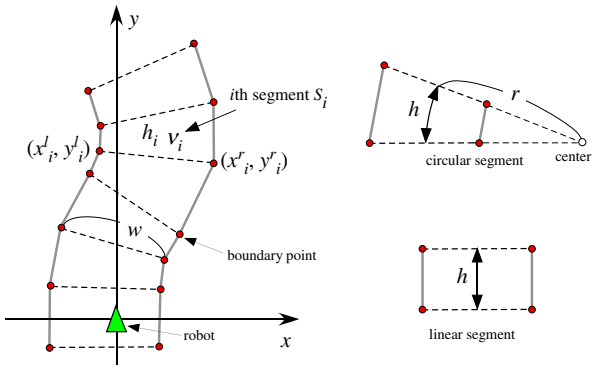


Fig. 2. A piecewise-linear road model (unbranched road model).

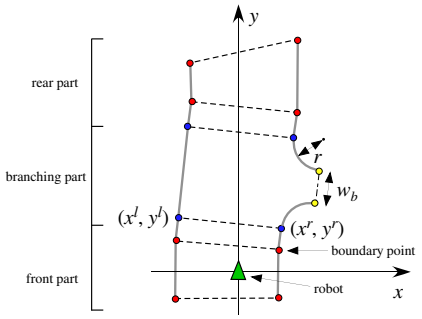


Fig. 3. A branching road model (right T-branch).

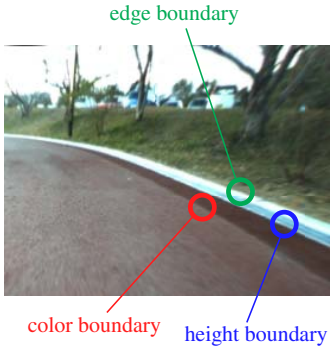


Fig. 4. Gaps between road boundary features.

thus explicitly estimate the gaps as a part of the boundary parameters. The position of road boundary is set at that of color boundary and the positional difference of edge or height boundary from there is modeled.

The gap G is thus represented by:

$$G = \left[g^r_e, g^l_e, g^r_h, g^l_h\right], \tag{1}$$

where superscripts mean the side (left or right) and subscripts mean the type (edge or height). In our previous method [12], only a gap between image and range boundary is estimated. The new gap model considers all gaps between boundaries. The effect of explicitly estimating the gaps will experimentally shown in Sec. III-C.

3) State vector: The unbranched road model is represented by:

$$X_u = [U, w, G, S_1, S_2, \ldots, S_6], \tag{2}$$
$$U = [\Delta x, \Delta y, \Delta\theta], \tag{3}$$
$$S_i = \left[x^r_i, y^r_i, x^l_i, y^l_i, \nu_i\right], \tag{4}$$

where U is the robot ego-motion, S_i is the parameters, which are two boundary points on both sides ((x^r_i, y^r_i) and (x^l_i, y^l_i)) and the local curvature ν_i, of the ith segment, w is the width of the road, respectively. w is assumed to be common to the current set of segments, but is estimated on-line. Length h_i is set to a constant (currently, $1.0\,[m]$).

The model for T-branches has extra parameters S_b for the branching part, represented by:

$$S_b = \left[x^l, y^l, x^r, y^r, w_b, r\right], \tag{5}$$

where (x^l, y^l) and (x^r, y^r) are the entry points, w_r is the width of the branch, r is the radius of the branching point. The branching road model is then given by:

$$X_b = \left[U, w, G, S^f_1, S^f_2, \ldots, S_b, S^r_1, S^r_2, \ldots\right], \tag{6}$$

where S^f_i and S^r_i are the segments for the front and the rear part; the number of these segments varies according to the width w_b of the branch.

Extra parameters S_c for the model of crossings is represented by:

$$S_c = \left[x^l, y^l, x^r, y^r, w^l_b, w^r_b, r^l, r^r\right], \tag{7}$$

where the width of branch and the radius are used for respective branches. The model for crossing is similar to that for T-branches (see eq. (6)).

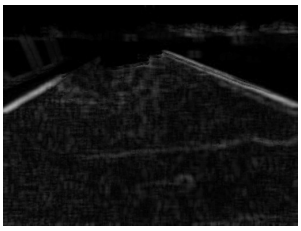
C. Image feature extraction and likelihood calculation

Evidence from three kinds of stereo image features (color, edge, and height) is extracted by generating three gradient images as shown in Fig. 5. A color gradient image is generated by first estimating the road region color model from a set of latest frames and then calculating gradient. An intensity gradient image is calculated by applying a median filter to the input image followed by a Sobel filter with Gaussian smoothing. A height gradient image is generated by first converting the input stereo depth image into a the height image, which represents the height of each pixel in the robot local coordinates, and then apply a differentiation with a Gaussian smoothing.

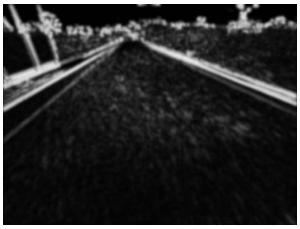
These gradient images are normalized to the range $[0, 255]$ and used for likelihood calculation as follows. The road model with respect to the robot pose is mapped onto the image and the gradient values under the mapped boundary are collected and averaged. The gaps between boundaries of different features are considered here in mapping. This averaged value is then transformed to a likelihood value ranging $[0, 1]$ using a sigmoid function:

$$L(x) = \frac{a}{1 + \exp\{-k(x - x_c)\}}, \tag{8}$$

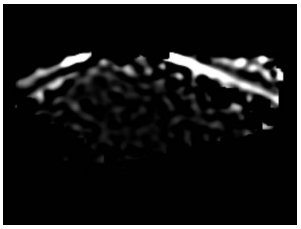
(a) input image



(b) color gradient image



(c) intensity gradient image



(d) height gradient image

Fig. 5. Input and gradient images for Road boundary features.

where parameters a, k, and x_c are determined empirically for each image feature.

Six likelihood values are calculated for every combination of the three features (color, edge, and height) and the two sides (left and right). The importance weight of a particle is given by the product of all likelihood values. In some cases, however, the likelihood values for a combination become very small *for any particles* due to, for example, a discontinuity of curb or strong cast shadows. This makes the weights for all particles become very small and many promising particles might be deleted. To avoid this, if the maximum likelihood for a feature on one side is less than a threshold (currently, 0.1), the combination of the feature and the side is considered not to be effective and is not used.

D. Road model update

The state transition step in particle filter transforms a set of particles to another set by robot ego-motion estimate and road model update. The former is calculated from visual odometry [12]. The latter is a key of our tracking method, which adaptively generates new particles to cope with road type changes. The road model update takes place when the robot is judged to enter a new road segment. The previous segment where the robot was is deleted and a new one is attached as shown in Figs. 6 and 7.

In the case of unbranched road, one usual road segment is attached. For each particle which should be updated, the curvature of the attached segment is chosen by sampling from a given probabilistic distribution.

The branching parts of a road gradually become visible as the robot moves, similarly to the case of unbranched roads. It is therefore possible to always make hypotheses of branching roads when new road segments are attached. Since the occurrence of branching parts is much smaller than ordinary road segments, however, such a hypothesis generation may waste particles. We thus add branching road

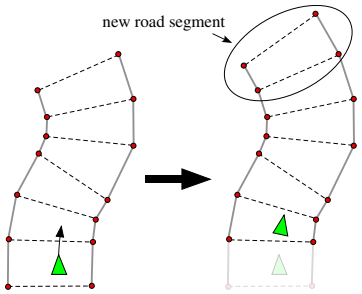


Fig. 6. Road model update.

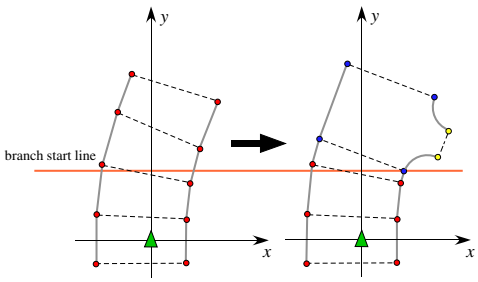


Fig. 7. Generating a branching road model.

models only when they are likely to be approaching.

For this purpose, we examine the *trends* of the likelihood values for the intensity gradient, the color gradient, and the height gradient along the direction of the road to determine if a branch or crossing is approaching. We calculate their averaged values for all particles and describe them as functions of the distance from the robot along the road. We try to find regions where all averaged values are less than some threshold (currently, 0.2) and whose length is larger than another threshold (currently, $1\,[m]$). If such a region exists, particles for appropriate branching models are generated.

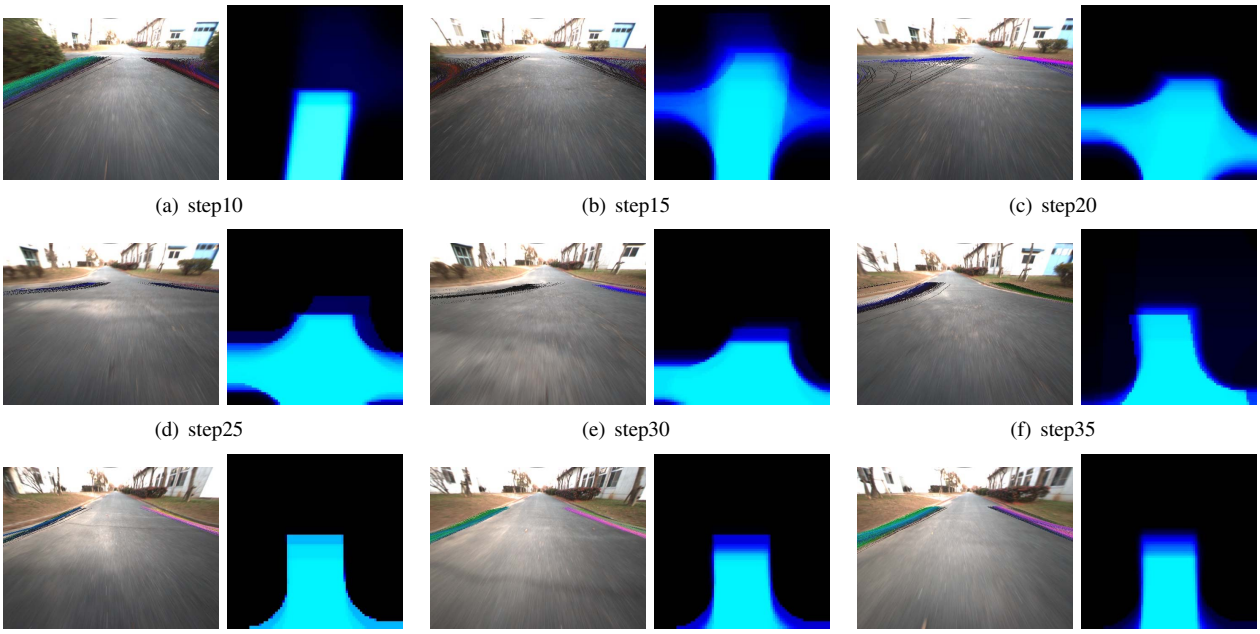
Fig. 8 shows an example of likelihood trend. Since there is a region with low likelihood values on the right, particles for right T-branches are generated.

III. Road boundary tracking experiments

This section describes results of off-line road boundary tracking using stored data. Fig. 9 shows two routes for the experiments. The first route is about $100\,[m]$ long and has two crossings. The second one is about $80\,[m]$ long; it is pavement with slight curvature and with roadside trees on one side.

A. Robot and sensor

Fig. 10 shows our mobile robot. The mobile base is an electric wheelchair, PatraFour by Kanto Auto Works Co., which can be controlled by a laptop PC (Core i7 620M, 2.66GHz, 3GB memory). We use a Bumblebee2 stereo camera by Point Grey Research Inc. ($100\,[deg.]$ horizontal field of view), set at the height of about $1.70\,[m]$ looking down by $25\,[deg.]$ for gaining maximum visibility of road surface. A sunshade is attached to the camera to prevent the sunlight.

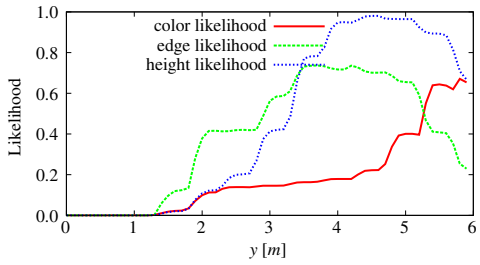


(a) step10 (b) step15 (c) step20

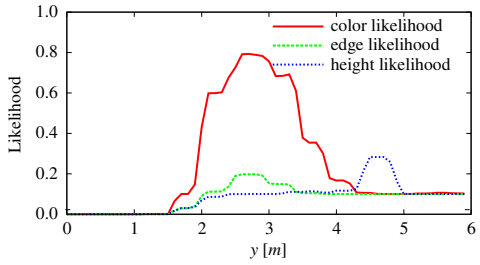
(d) step25 (e) step30 (f) step35

(g) step40 (h) step45 (i) step50

Fig. 11. Results of road boundary modeling.



(a) left boundary



(b) right boundary

Fig. 8. An example of likelihood trend

(a) First route. (b) Second route.

Fig. 9. Two routes for off-line experiments.

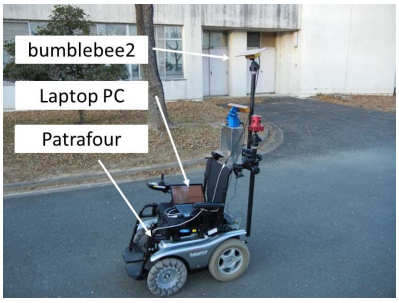


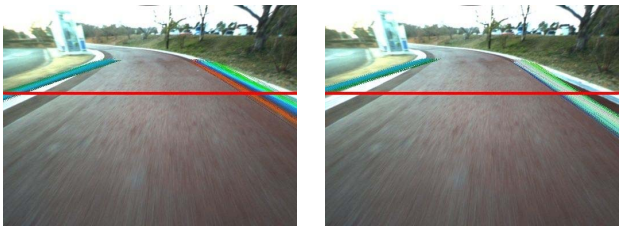
Fig. 10. Our mobile robot and a stereo camera.

B. Road boundary tracking results

Fig. 11 shows tracking results for the first route. The left image in each figure indicates road boundaries obtained from the particle set superimposed on the input image. To see which feature is effective, we assign the three primary colors, red, green, and blue, to color, edge, and height information, respectively. Mixture of colors is obtained when multiple features are effective. The right image in each figure is a kind of certainty distribution of road regions in the robot local coordinates, obtained by voting road regions coming from the current set of particles. Brighter pixels indicate higher certainties. We can see road shape is correctly recognized in various scenes including crossings.

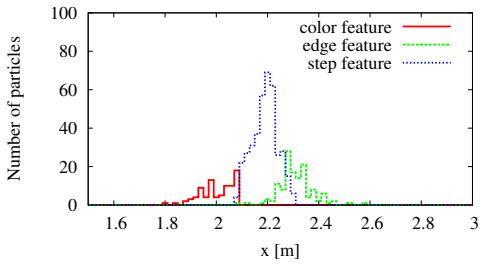
C. Effect of estimating gap between image features

Estimating gaps between every image feature is an extension of the method proposed in this paper. We here examine the effect of gap estimation. Fig. 12 shows the difference of estimated road boundaries for the cases with and without gap estimation. In the gap estimation case, the position of

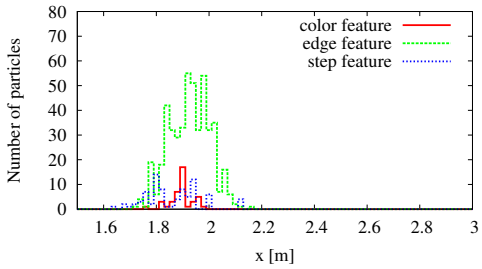


(a) without gap estimation (b) with gap estimation

Fig. 12. Comparison of estimation with and without gap estimation.



(a) without gap estimation



(b) with gap estimation

Fig. 13. Distributions of road boundary position.

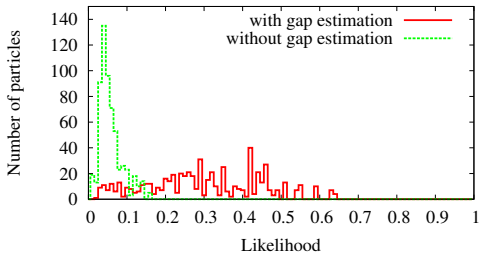


Fig. 14. Comparison of likelihood distributions.

the road boundary is shown at that of color features. White colors of superimposed boundaries indicate multiple features are more effectively integrated than in the case without gap estimation.

For a more precise examination, we choose the position $2.5\,[m]$ ahead of the robot and calculate the distribution of boundary positions and likelihoods of particles. The red line in Fig. 12 indicates the position for examination.

Fig. 13 shows the distributions of the horizontal positions on the red line. We divide the particles into three groups by the most dominant (i.e., the highest likelihood) feature. Fig. 14 shows the distribution of the total likelihood of particle. These figures show that with the gap estimation, particles are distributed in a narrower area and the total likelihood is

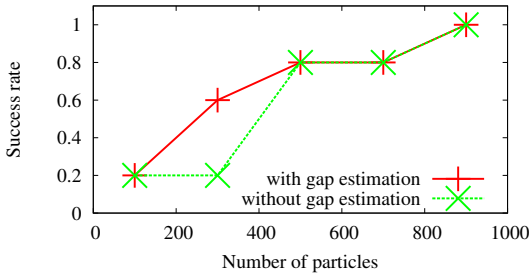


Fig. 15. Comparison of success rates.

higher because the pieces of evidence from different features at different positions are effectively integrated.

Fig. 15 compares success rates for the two cases. We ran the tracking method five times for each case and calculated the success rates. A run is considered successful if the boundary tracking is judged to be correct until the end by human visual inspection of the resultant road boundaries mapped onto the image sequence. The figure shows that the necessary number of particles, that is, the calculation cost can be made smaller by the gap estimation while keeping the reliability of estimation.

IV. Navigation Experiments

Autonomous driving experiments were conducted at the routes shown in Fig. 16. The robot moved at $0.7\,[m/s]$ and the frame rate was about $3\,[fps]$ with the number of particles being 500. Based on the recognition result of road shape, the robot was controlled to move at the center of the road.

There are four crossings in the fourth route. A crossing is considered recognized when the ratio of particles corresponding to the crossing road type exceeds a threshold (currently, 0.8). The shape of a crossing is calculated by the weighted sum of the crossing particles. The turning direction at each crossing is manually given and the robot moves autonomously by dead reckoning using odometry inside the crossing.

Fig. 17 shows the results of autonomous driving in the third and the fourth route. The robot moved successfully in a variety of road scenes and conditions.

V. Conclusions and Discussion

This paper has described a road boundary tracking method using only one stereo camera. We adopt and extend our previous method which uses flexible road models and a particle filter. We conducted boundary tracking and autonomous driving experiments in four different places in our campus and verified the robustness of the proposed method to the variety of road scene and conditions. We have also shown the effectiveness of explicit estimation of gaps between image features supporting road boundaries.

Currently we have one unbranched road model and three branching road models. This is certainly not enough considering a great variety of road scenes. For so-called branching roads, which can be represented by a branching part and a set of branches, however, our model can relatively easily be

(a) Third route. (b) Fourth route.

Fig. 16. Two routes for autonomous driving experiments.

(a) Third route.

(b) Fourth route.

Fig. 17. Autonomous driving at the third and the fourth route in Fig. 16.

extended to cover such branching roads, although it is still necessary to devise a procedure for observing *trends* and generating appropriate particles.

On the other hand, different road (or space) representation will be necessary for open spaces where several objects or boundaries determine the traversable regions; an example is a space around a fountain in a park. Current road models assume that a pair of left and right boundary are locally in parallel. Such a parallelism will no longer exist for open spaces. One way would be to model the pieces of boundaries separately; we are now working on developing new boundary models.

As mentioned in Introduction, several functions are necessary for realizing fully autonomous navigation. The proposed method, combined with a road following control, has made it possible to move the robot along a path. By additionally using some obstacle detection and avoidance capability, the robot can move around safely, as people do. Another important function is the global localization and navigation, that is, determining the robot position and the direction to move. GPS-based systems (e.g., [14]) or view-based systems (e.g., [15]) will be candidates for providing such a function. We are now working on combining the proposed road boundary tracking with our view-based localization method [16].

Acknowledgment

This work is supported in part by Grant-in-Aid for Scientific Research (No. 21300075) from JSPS.

REFERENCES

[1] M. Beauvais and S. Lakshmanan, "Clark: a heterogeneous sensor fusion method for finding lanes and obstacles," *Image and Vision Computing*, vol. 18, pp. 397–413, 2000.

[2] E. Dickmanns and B. Mysliwetz, "Recursive 3-d road and relative ego-state recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 199–213, 1992.

[3] J. Crisman and C. Thorpe, "Scarf: A color vision system that tracks roads and intersections," *IEEE Trans. on Robotics and Automat.*, vol. 9, no. 1, pp. 49–58, 1993.

[4] M. Sotelo, F. Rodriguez, L. Magdalena, L. Bergasa, and L. Boquete, "A color vision-based lane tracking system for autonomous driving on unmarked roads," *Autonomous Robots*, vol. 16, pp. 95–116, 2004.

[5] M. Okutomi, K. Nakano, J. Matsuyama, and T. Hara, "Robust estimation of planar regions for visual navigation using sequential stereo images," in *Proceedings of 2002 IEEE Int. Conf. on Robotics and Automation*, 2002, pp. 3321–3327.

[6] H. Cramer and G. Wanielik, "Road border detection and tracking in non cooperative areas with a laser radar system," in *Proceedings of German Radar Symposium*, 2002.

[7] W. Wijesoma, K. Kodagoda, and A. Balasuriya, "Road boundary detection and tracking using ladar sensing," *IEEE Trans. on Robotics and Automation*, vol. 20, no. 3, pp. 456–464, 2004.

[8] S. Thrun and et al., "Stanley: The robot that won the darpa grand challenge," *J. of Field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[9] N. Apostoloff and A. Zelinsky, "Robust vision based lane tracking using multiple cues and particle filtering," in *Proceedings of 2003 IEEE Intelligent Vehicles*, 2003, pp. 558–563.

[10] R. Danescu and S. Nedevschi, "Probabilistic lane tracking in difficult road scenarios using stereovision," *IEEE Trans. on Intelligent Transportation Systems*, vol. 10, no. 2, pp. 272–282, 2009.

[11] P. Santana, N. Alves, L. Correia, and J. Barata, "Swarm-based visual saliency for trail detection," in *Proceedings of 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010, pp. 759–765.

[12] Y. Matsushita and J. Miura, "On-line road boundary modeling with multiple sensory features, flexible road model, and particle filter," *Robotics and Autonomous Systems*, vol. 59, no. 5, pp. 274–284, 2011.

[13] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. The MIT Press, 2005.

[14] S. Panzieri, F. Pascucci, and G. Ulivi, "An outdoor navigation system using gps and inertial platform," in *Proceedings of 2001 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, vol. 2, 2001, pp. 1346–1351.

[15] M. Cummins and P. Newman, "Fab-map: Probabilistic localization and mapping in the space of appearance," *Int. J. of Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.

[16] J. Miura and K. Yamamoto, "Robust view matching-based markov localization in outdoor environments," in *Proceedings of 2008 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2008, pp. 2970–2976.