

Texture-Based Vanishing Point Voting for Road Shape Estimation

Christopher Rasmussen

Dept. Computer & Information Sciences
University of Delaware
cer@cis.udel.edu

Abstract

Many rural roads lack sharp, smoothly curving edges and a homogeneous surface appearance, hampering traditional vision-based road-following methods. However, they often have strong texture cues parallel to the road direction in the form of ruts and tracks left by other vehicles. This paper describes an unsupervised algorithm for following ill-structured roads in which dominant texture orientations computed with Gabor wavelet filters vote for a consensus road vanishing point location. The technique is first described for estimating the direction of straight-road segments, then extended to curved and undulating roads by tracking the vanishing point indicated by a differential “strip” of voters moving up toward the nominal vanishing line. Finally, the vanishing point is used to constrain a search for the road boundaries by maximizing texture- and color-based region discriminant functions. Results are shown for a variety of road scenes including gravel roads, dirt trails, and highways.

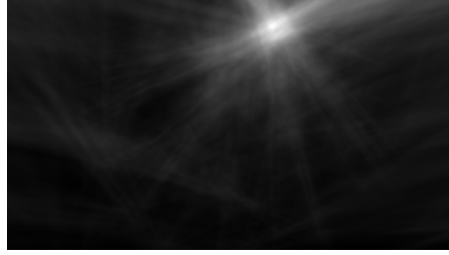
1. Introduction

Many complementary strategies for visual road following have been developed based on certain assumptions about the characteristics of the road scene. For example, edge-based methods such as those described in [17, 16, 1] are often used to identify lane lines or road borders, which are fit to a model of the road curvature, width, and so on. These algorithms typically work best on well-engineered roads such as highways which are paved and/or painted, resulting in a wealth of high-contrast contours suited for edge detection. Another popular set of methods for road tracking are region-based [1, 3, 10, 20]. These approaches use characteristics such as color or texture measured over local neighborhoods in order to formulate and threshold on a likelihood that pixels belong to the road area vs. the background. When there is a good contrast for the cue chosen, there is no need for the presence of sharp or unbroken edges, which tends to make these methods more appropriate for unpaved rural roads.

Most road images can be successfully interpreted using a variant of one of the two above approaches. Nonetheless, there are some scenes that possess neither strong edges nor contrasting local characteristics. Figure 1(a) shows one such road (the cross was added by our algorithm and is explained in Section 2.1). It is from



(a)



(b)

Figure 1: (a) Desert road from DARPA Grand Challenge example set (with vanishing point computed as in Section 2.1); (b) Votes for vanishing point candidates (top 3/4 of image)

a set of “course examples” made available to entrants in the 2004 U.S. DARPA Grand Challenge, an autonomous cross-country driving competition [4] [DARPA air-brushed part of the image near the horizon to obscure location-identifying features]. There is no color difference between the road surface and off-road areas and no strong edges delimiting it. The one characteristic that seems to define such roads is texture, but not so much in a locally measurable sense, because there are bumps, shadows, and stripes everywhere. Rather, one seems to apprehend the roads most easily because of their overall *banding* patterns. This banding, presumably due to ruts and tire tracks left by previous vehicles driven by humans who knew the way, is aligned with the road direction and thus most apparent because of the strong grouping cue imposed by its vanishing point.

A number of researchers have used vanishing points as global constraints for road following or identification of painted features on roads (such as so-called “zebra crossings,” or crosswalks) [14, 13, 15, 9, 2]. Broadly, the key to the approach is to use a voting procedure like a Hough transform on edge-detected line segments to find points where many intersect. Peaks in the voting function are good candidates for vanishing points. This is sometimes called a “cascaded” Hough transform [19] because the lines themselves may have first been identified via a Hough transform. Similar grouping strategies have also been investigated outside the context of roads, such as in urban and indoor environments rich in straight lines, in conjunction with a more general analysis of repeated elements and patterns viewed under perspective [18, 5].

All of the voting methods for localizing a road’s vanishing point that we have identified in the literature appear to be based on a prior step of finding line segments via edge detection. Moreover, with the exception of [9], vanishing-point-centric algorithms appear not to deal explicitly with the issue of road curvature and/or undulation, which remove the possibility of a unique vanishing point associated with the road direction. Both of these limitations are problematic if vanishing point methods are to be applied to bumpy back-country roads like the

desert scene discussed above.

In this paper we present a straightforward method for locating the road’s vanishing point in such difficult scenes through texture analysis. Specifically, we replace the edge-detection step, which does not work on many such images because the road bands are too low-frequency to be detected, with estimates of the *dominant orientation* at each location in the image. These suffice to conduct voting in a similar fashion and find a vanishing point.

We also add a second important step to deal with road curvature (both in- and out-of-plane), which is the notion of tracking the vanishing points associated with differential segments of the road as they are traced from the viewer into the distance. By integrating this sequence of directions, we can recover shape information about the road ahead to aid in driving control. The method is easily extended to temporal tracking of the vanishing point over sequences of images.

Finally, the estimated road curvature alone does not provide information about the vehicle’s lateral displacement that would allow centering—for this we need estimates of the left and right road boundaries. Vanishing point information provides a powerful constraint on where these edges might be, however, by defining a family of possible edge lines (for straight roads) and edge contours (for curved roads) radiating outward (below the horizon line) from a single image location. This allows the road segmentation task to be formulated as simply a 2-D search among these curves for left and right boundaries which maximize the difference of some visual discriminant function inside the road region vs. outside it. In this paper, we describe an approach to combining measures of texturedness and color to robustly locate these edges.

2. Methods

There are four significant components to the road following algorithm. First, a dominant texture orientation $\theta(\mathbf{p})$ (the direction that describes the strongest local parallel structure or texture flow) is computed at every image pixel $\mathbf{p} = (x, y)$. Second, assuming a straight, planar road, all dominant orientations in the image vote for a single best road vanishing point. Third, if the road curves or undulates a series of vanishing points for tangent directions along the road must be estimated (for a sequence of road images, the deformation of this vanishing point contour from image to image must also be estimated, but due to space considerations it is not described in this paper). Finally, the left and right edges of the road are determined by optimizing a discriminant function relating visual characteristics inside and outside the road region. We describe the last three steps in the following subsections; our method of dominant orientation estimation using $n = 72$ Gabor wavelet filters [7] at equally spaced angles is explained in detail in [11].

2.1. Vanishing Point Voting

For a straight road segment on planar ground, there is a unique vanishing point associated with the dominant orientations of the pixels belonging to the road. Curved segments induce a set of vanishing points (discussed below). Though road vanishing points may lie outside the field of view (FOV) of the road-following camera, it is reasonable to limit our search to the area of the image itself if the

following conditions are met: (1) The camera’s optical axis heading and tilt are aligned with the vehicle’s direction of travel and approximately level, respectively; (2) Its FOV is sufficiently wide to accommodate the maximum curvature of the road; and (3) The vehicle is roughly aligned with the road (i.e., road following is proceeding successfully).

Furthermore, we assert that for most road scenes, especially rural ones, the vanishing point due to the road is the only one in the image. In rural scenes, there is very little other coherent parallel structure besides that due to the road. The dominant orientations of much off-road texture such as vegetation, rocks, etc. are randomly and uniformly distributed with no strong points of convergence. Even in urban scenes with non-road parallel structure, such texture is predominantly horizontal and vertical, and hence the associated vanishing points are located well outside the image.

In the following subsections we describe methods for formulating an objective function $votes(\mathbf{v})$ to evaluate the support of road vanishing point candidates $\mathbf{v} = (x, y)$ over a search region C roughly the size of the image itself, and how to efficiently find the global maximum of $votes$.

2.1.1 Straight Roads

The possible vanishing points for an image pixel \mathbf{p} with dominant orientation $\theta(\mathbf{p})$ are all of the points (x, y) along the line defined by $(\mathbf{p}, \theta(\mathbf{p}))$. Because our method of computing the dominant orientations has a finite angular resolution of $\frac{n}{\pi}$, uncertainty about the “true” $\theta(\mathbf{p})$ should spread this support over an angular interval. Thus, if the angle of the line joining an image pixel \mathbf{p} and a vanishing point candidate \mathbf{v} is $\alpha(\mathbf{p}, \mathbf{v})$, we say that \mathbf{p} votes for \mathbf{v} if the difference between $\alpha(\mathbf{p}, \mathbf{v})$ and $\theta(\mathbf{p})$ is within the dominant orientation estimator’s angular resolution. Formally, this defines a voting function as follows:

$$vote(\mathbf{p}, \mathbf{v}) = \begin{cases} 1 & \text{if } |\alpha(\mathbf{p}, \mathbf{v}) - \theta(\mathbf{p})| \leq \frac{n}{2\pi} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

This leads to a straightforward objective function for a given vanishing point candidate \mathbf{v} :

$$votes(\mathbf{v}) = \sum_{\mathbf{p} \in R(\mathbf{v})} vote(\mathbf{p}, \mathbf{v}) \quad (2)$$

where $R(\mathbf{v})$ defines a *voting region*. For straight, flat roads, we set $R(\mathbf{v})$ to be the entire image, minus edge pixels excluded from convolution by the kernel size, and minus pixels *above* the current candidate \mathbf{v} . Only pixels below the vanishing line \mathbf{l} implied by \mathbf{v} are allowed to vote because support is only sought from features in the plane of the road (though in urban scenes out-of-plane building features, etc. may corroborate this decision [15]).

In this work, we assume that \mathbf{l} is approximately horizontal. Other researchers have inferred \mathbf{l} for road and urban scenes by computing a second vanishing point obtained with either another set of parallel lines [19, 13, 12] or the cross ratio of equally spaced lines [13]. The road scenes we are considering have insufficient structure to carry out such computations.

An example of *votes* computed at a 1-pixel resolution over $C =$ the top three-fourths of the image in Figure 1(a) (where the maximum vote-getter \mathbf{v}_{max} is indicated with a cross) is shown in Figure 1(b).

An exhaustive search over C for the global maximum of *votes* of the type necessary to generate Figure 1(b) is costly and unnecessary. Rather, a hierarchical scheme is indicated in order to limit the number of evaluations of *votes* and control the precision with which \mathbf{v}_{max} is localized. Because it integrates easily with the tracking methods described in the next subsection, we perform a randomized search by carrying out a few iterations of a particle filter [6] initialized to a uniform, grid-like distribution over C (with spacing $\Delta x = \Delta y = 10$).

2.1.2 Curved Roads

Road curvature and/or non-planarity result in a set of different apparent vanishing points associated with different tangents along the section of road seen by the camera. Therefore the procedure of the preceding subsection, which assumes a unique vanishing point, must be modified to estimate the sequence of vanishing points which correspond to differential segments of road at increasing distances from the camera. Furthermore, over a sequence of images gathered as the camera moves along the road, the *vanishing point contour* thus traced for a single image must itself be tracked from frame to frame.

Suppose the spine of the approaching road section that is visible to the camera is parametrically defined by a nonlinear space curve $\mathbf{x}(u)$, with increasing u indicating greater distance along the road. If $\mathbf{x}(u)$ lies entirely in one plane then the image of these vanishing points $\mathbf{v}(u)$ is a 1-D curve on the vanishing line \mathbf{l} . If $\mathbf{x}(u)$ is not in the plane, then the vanishing line varies with distance according to $\mathbf{l}(u)$ (still assumed to be horizontal) and $\mathbf{v}(u)$ is a 2-D curve.

We cannot directly recover $\mathbf{v}(u)$ since $\mathbf{x}(u)$ is unknown. However, u is a monotonically increasing function of the image scanline s , where $s = 0$ is the bottom row of pixels, so we can attempt to estimate the closely related curve $\mathbf{v}(s)$. This implies modifying Equation 2 to

$$votes(\mathbf{v}(s)) = \sum_{\mathbf{p} \in R(\mathbf{v}, s \pm \Delta)} vote(\mathbf{p}, \mathbf{v}) \quad (3)$$

where $R(\mathbf{v}, s \pm \Delta)$ is now a differential horizontal *strip* of voting pixels centered on scanline s . Smaller values of the height of the strip 2Δ yield a more accurate but less precise approximation of the road tangent ($\Delta \approx 0.1h$, where h is the image height, for the results in this paper). s is iterated from 0 until $\mathbf{v}_{max}(s) \in R(\mathbf{v}, s \pm \Delta)$ (roughly the point where the strip crosses the vanishing line).

A strip-based approach to vanishing point detection for curvature estimation was also used in [9] for edge-detected road boundaries and lane lines, but with only a few non-overlapping strips.

Furthermore, we do not simply estimate a best vanishing point fit $\mathbf{v}_{max}(s)$ for every s independently by rerunning the full randomized search over C as described in the previous subsection. Rather, we *track* the vanishing point by continuing to run the particle filter with weak dynamics $p(\mathbf{v}(s) | \mathbf{v}(s-1))$ (e.g., a low-variance, circular Gaussian). This allows a more accurate estimate of $\mathbf{v}_{max}(s)$ because of

the concentration of particles already in the solution area, and reduces the chance of misidentification of the vanishing point due to a false peak somewhere else in the image.

The vanishing point of each strip s implies a tangent to the image of the road curve at s . By hypothesizing an arbitrary point on the road, we can integrate this tangent function over s (i.e., with Euler steps) and thereby trace a curve or “flow line” followed by the road point. The results of this process are illustrated in Figure 4 for two example road segments.

2.2. Constrained Road Segmentation

Searching for the road boundaries *de novo* would be onerous, but knowledge of the vanishing point location(s) (plural for curved roads) provides a powerful constraint on where they may be. Regardless of how curvy the road is, the flow lines induced by the vanishing point tracking process represent a 1-D family of possible road edges. Thus, the search for an optimal pair of left and right edges (l, r) is only 2-dimensional, and fairly tightly bounded by the range of possible road widths and the vehicle’s maximum expected lateral departure from the road midline.

The criterion $Q(l, r)$ which we seek to optimize is the difference between the average values of some characteristic J within the image road region $road(l, r)$ (as defined by the candidate left and right edge curves and the borders of the image) and that characteristic in the region outside the road $offroad(l, r)$ (see [8] for a related approach applied to autonomous harvesting). For MAP estimation, we combined this with a weak Gaussian prior on edge locations $p(l, r)$ to obtain the following expression:

$$Q(l, r) = |\bar{J}(road(l, r)) - \bar{J}(offroad(l, r))| \cdot p(l, r) \quad (4)$$

We used four region-based cues to distinguish the road: the color channels J_R , J_G , and J_B ; and a measure of texturedness J_T . Because it was already computed to obtain dominant orientations, it was inexpensive to use the magnitude of the maximum complex response $J_T = \mathbf{I}_{complex}(x, y)$ at each pixel ¹.

The objective functions Q_T for texturedness, Q_R for redness, Q_G for greenness, and Q_B for blueness are shown in the lower part of Figure 2 for an example straight road image (upper left) and its computed vanishing point. The possible angles $\theta(l)$ and $\theta(r)$ of the straight candidate road edges with respect to the horizontal vanishing line were limited to $n = 72$ discrete values. Thus the Q functions are plotted as 72×72 height maps, with each row representing a hypothetical left edge and each column a hypothetical right edge.

The computed road edges for each cue corresponding to the maximum of its objective function, indicated with a cross in the height map, are plotted as lines on the input image. Each of the color channel cues localizes the left road edge well,

¹We have experimented with directly using the distribution of pixels J_V that voted for \mathbf{v}_{max} , because by our assumptions the density of such voters should be much heavier within the road than outside it. For heavily banded roads, this approach was successful, but it did not extend to lightly-textured roads with strong edges (like paved roads with painted lanes, for which our vanishing point localization method works quite well). The key problem is that voters for such roads are concentrated along their edges rather than distributed throughout the interior of the road region.

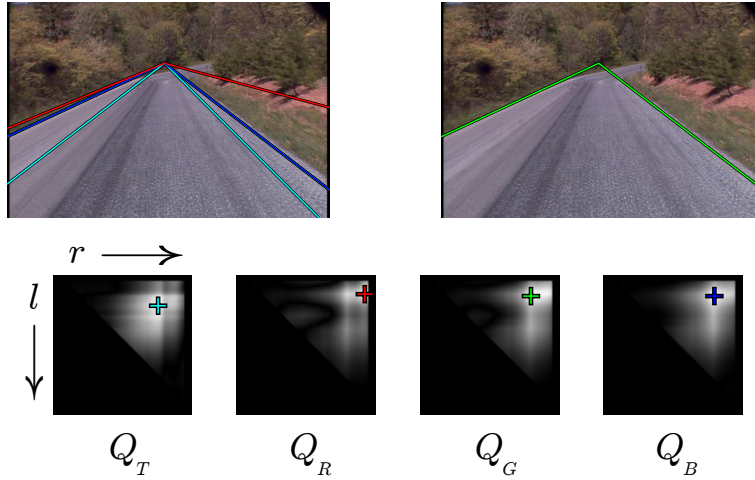


Figure 2: Road segmentation example: (Top left): Image with best texture-based edges (inner cyan lines), color-based edges (outer red, blue, green); (Top right): Best left and right edge estimates from green cue; (Bottom): Underlying objective functions Q for texture and color channels.

while the right edge is found correctly only by the blue and green cues (the green line is overdrawn by the blue). The red cue has a bimodal distribution on the right edge and chooses the outer, incorrect option, while the texture cue's estimate errs inside the road on both sides.

Empirically, we have found that the left and right road edges are well-localized over a diverse set of straight-road images in one of two ways: (1) One or more cues find both edges accurately; or (2) One or more cues find only one edge accurately, while another one or more cues localize the other edge well. Which cue or cues exhibit good performance depends on the characteristics of the road material, lighting conditions, and the distribution of flora and rocks in the off-road image regions. Situation (2) most often occurs because the background has different characteristics on the left and right sides of the road.

We have gotten promising preliminary results based on a simple measure of the peakedness of the Q function. Specifically, we rank every cue a 's left edge estimate \hat{l}_a and right edge estimate \hat{r}_a separately by calculating the kurtosis of the 1-D functions $Q_a(l, \hat{r}_a)$ and $Q_a(\hat{l}, r)$, respectively, with the highest winning. The intuition behind this measure is that peakedness is correlated with confidence in the solution, and all else being equal we want a left edge estimate that is clearly above the other choices given the right edge estimate, and vice versa. The best left and right edges in the example image were both found by the green cue using this procedure, as shown in the upper right of Figure 2.

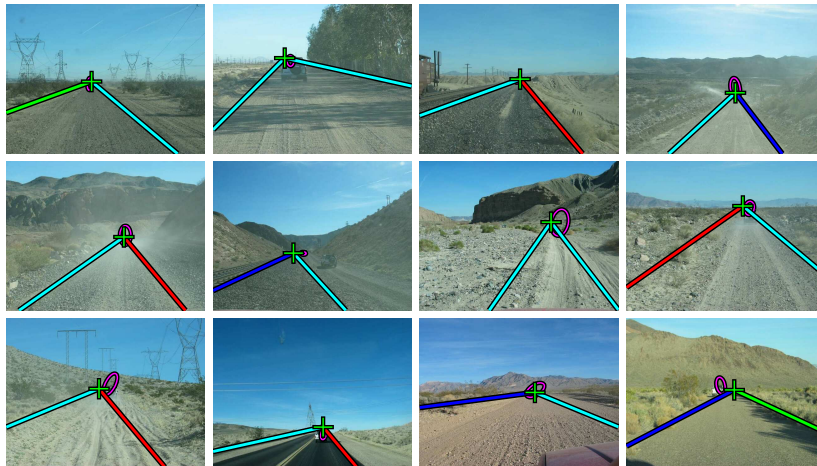


Figure 3: Computed vanishing points are shown as crosses, with Gaussian fits of a set of human responses marked with ellipses

3. Results

Straight roads Vanishing point localization by dominant texture orientation voting worked robustly on predominantly straight roads with a wide variety of surface characteristics. Some results are shown in Figure 3. 16 illustrative images (the “Mojave” data) were chosen from a large set of high-resolution digital photographs taken on a scouting trip along a possible Grand Challenge route in the Southern California desert. The algorithm was run on resampled 160×120 versions of the images using particle-based search; the figure shows the computed \mathbf{v}_{max} for 12 of the 16 images with a green cross. To assess the algorithm’s performance vs. human perception of the vanishing point location, we conducted a small web-based study: ~ 30 people were given a short definition of road vanishing points, shown two different example images with the vanishing point marked, and asked to click where they thought the vanishing point was in 640×480 versions of each of the 16 Mojave images. 16 subjects completed the study; 11 of their 256 choices (4.3%) were manually removed as obvious misclick outliers. The figure indicates the distribution of human choices with purple 3σ error ellipses, most of which were fairly tight. The median positional difference at the 320×240 scale between our algorithm’s estimates and the human choices was 6.2 pixels horizontally and 4.3 pixels vertically.

Curved roads Examples of tracking the vanishing point for curved, non-planar roads from a set of 720×480 DV camera images captured on a variety of roads at Fort Indiantown Gap, PA are shown in Figures 4(a) and (b). The yellow contour indicates the trace of the estimated vanishing point positions as the scanline s was incremented and the set of voters changed to points farther along the road. Horizontal movement of the vanishing point is of course proportional to left-right

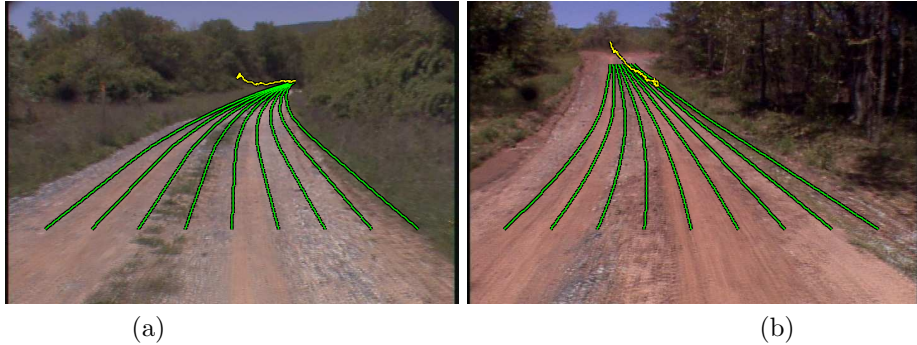


Figure 4: Tracks of vanishing point (yellow) for curved road and induced road “flow lines” (green)

road curvature, and vertical movement indicates a rise or dip in the road. Thus we see that the road in Figure 4(a) is relatively level with a slight dip in the middle, which is correct. On the other hand, the vanishing point for the road in Figure 4(b) rises, indicating an approaching hill.

A more intuitive interpretation of the vanishing point motion can be obtained by simply integrating the implied tangent to the image of the road curve (the line joining any point on scanline s with its vanishing point $\mathbf{v}(s)$) for a few sample points. These are the green contours, or “flow lines,” in the images in Figures 4. These curves are not meant to indicate the width of the road, for no segmentation has been carried out, but rather only to illustrate the structure of the terrain that the road ahead traverses.

Segmentation The vanishing points found automatically in Figure 3 were used to constrain a search for the road edges in those images. The best left and right edges obtained using the method of Section 2.3 are overlaid on the input images, with the color of each side indicating which cue’s estimate was used (cyan = J_T , with the other colors as expected). For efficiency, only $n = 72$ possible edge locations angles relative to the vanishing point location were tested, limiting the angular resolution of road boundary localization. Nonetheless, the performance was strong on a variety of low-contrast roads, a number of which have ambiguous borders even to a human (e.g., row 3, column 1–(3, 1); or the railroad embankments in (1, 3) and (2, 2)). Overestimates of the road width, which are perhaps most dangerous to an autonomous vehicle, are relatively small, and never so large that the midline is off the road.

4. Conclusion

We have presented an algorithm for road following that at its most basic level relies on road texture to identify a global road vanishing point for steering control. By spatially partitioning the voting procedure into strips parallel to the vanishing line and tracking the results, we can estimate a sequence of vanishing points for curved road and recover road curvature through integration. Finally, we have

leveraged the information provided by the vanishing point to make the problem of road boundary identification highly manageable even on road scenes with very little structure. With no training, the system is robust to a variety of road surface materials and geometries, and it runs quickly enough for real vehicle control.

Though our method for road segmentation performs satisfactorily on many scenes, the general issue of how to identify the best cue or cues for a given image is a difficult model selection problem, and needs to be examined in more detail.

5. Acknowledgments

Portions of this work were supported by a grant from the National Institute of Standards & Technology. The Mojave data was generously provided by Caltech's Grand Challenge team.

References

- [1] N. Apostoloff and A. Zelinsky. Robust vision based lane tracking using multiple cues and particle filtering. In *Proc. IEEE Intelligent Vehicles Symposium*, 2003.
- [2] J. Coughlan and A. Yuille. Manhattan world: Orientation and outlier detection by bayesian inference. *Neural Computation*, 15(5):1063–088, 2003.
- [3] J. Crisman and C. Thorpe. UNSCARF, a color vision system for the detection of unstructured roads. In *Proc. Int. Conf. Robotics and Automation*, pp. 2496–2501, 1991.
- [4] Defense Advanced Research Projects Agency (DARPA). DARPA Grand Challenge. Available at <http://www.darpa.mil/grandchallenge>. Accessed July 22, 2003.
- [5] A. Zisserman F. Schaffalitzky. Planar grouping for automatic detection of vanishing lines and points. *Image and Vision Computing*, 9(18):647–658, 2000.
- [6] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. European Conf. Computer Vision*, pp. 343–356, 1996.
- [7] T. Lee. Image representation using 2D Gabor wavelets. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(10):959–971, 1996.
- [8] M. Ollis and A. Stentz. Vision-based perception for an autonomous harvester. In *Proc. Int. Conf. Intelligent Robots and Systems*, pp. 1838–1844, 1997.
- [9] A. Polk and R. Jain. A parallel architecture for curvature-based road scene classification. In I. Masaki, editor, *Vision-Based Vehicle Guidance*, pp. 284–299. Springer, 1992.
- [10] C. Rasmussen. Combining laser range, color, and texture cues for autonomous road following. In *Proc. Int. Conf. Robotics and Automation*, 2002.
- [11] C. Rasmussen. Grouping dominant orientations for ill-structured road following. In *Proc. Computer Vision and Pattern Recognition*, 2004.
- [12] E. Ribeiro and E. Hancock. Perspective pose from spectral voting. In *Proc. Computer Vision and Pattern Recognition*, 2000.
- [13] S. Se. Zebra-crossing detection for the partially sighted. In *Proc. Computer Vision and Pattern Recognition*, pp. 211–217, 2000.
- [14] S. Se and M. Brady. Vision-based detection of staircases. In *Proc. Asian Conf. Computer Vision*, pp. 535–540, 2000.
- [15] N. Simond and P. Rives. Homography from a vanishing point in urban scenes. In *Proc. Int. Conf. Intelligent Robots and Systems*, 2003.
- [16] B. Southall and C. Taylor. Stochastic road shape estimation. In *Proc. Int. Conf. Computer Vision*, pp. 206–212, 2001.
- [17] C. Taylor, J. Malik, and J. Weber. A real-time approach to stereopsis and lane-finding. In *Proc. IEEE Intelligent Vehicles Symposium*, 1996.
- [18] A. Turina, T. Tuytelaars, and L. Van Gool. Efficient grouping under perspective skew. In *Proc. Computer Vision and Pattern Recognition*, 2001.
- [19] T. Tuytelaars, M. Proesmans, and L. Van Gool. The cascaded Hough transform. In *Proc. IEEE Int. Conf. on Image Processing*, 1998.
- [20] J. Zhang and H. Nagel. Texture-based segmentation of road images. In *Proc. IEEE Intelligent Vehicles Symposium*, 1994.