Proceedings of the 2003 IEEE/RSJ
Intl. Conference on Intelligent Robots and Systems
Las Vegas, Nevada · October 2003

Tracking Road Edges in the Panospheric Image Plane

Peter I. Corke[1], Dimitris Symeonidis[1] & Kane Usher[1,2]
[1]CSIRO Manufacturing & Infrastructure Technology
PO Box 883, Kenmore, Qld 4069, Australia
[2]Queensland University of Technology
2 George Street, Brisbane, Qld 4000, Australia
email: Peter.Corke@csiro.au

Abstract— **This paper presents a technique for tracking road edges in a panoramic image sequence. The major contribution is that instead of unwarping the image to find parallel lines representing the road edges, we choose to warp the parallel groundplane lines into the image plane of the equiangular panospheric camera (see video). Updating the parameters of the line thus involves searching a very small number of pixels in the panoramic image, requiring considerably less computation than unwarping. Results using real-world images, including shadows, intersections and curves, are presented.**

I. Introduction

We are interested in the control of mobile vehicles in semi-structured outdoor environments. In particular we are interested in navigation using vision as much as possible, since in environments such as building construction sites, GPS may be unreliable. We also wish to work as much as possible with the natural features of the environment, rather than instrumenting it with markers or beacons.

Our approach is to establish a number of fundamental behaviours or competencies which are required for useful outdoor tasks. Previous work has focused on the problem of vision-based homing [1] which will guide a mobile robot to a particular location in a workspace based on appearance. This paper is concerned with the problem of driving along a roadway, using information only from a panospheric camera.

The novel aspect of the work described in this paper is that the panospheric image is not unwarped. Instead we derive the equation of a curve in the image plane that corresponds to a line in the groundplane. This curve has two parameters: the angle of the line with respect to the vehicle's heading direction, and its horizontal offset from the line. An iterative method is described to adjust these parameters so as to bring the model curve and the observed curve into alignment, giving the distance and orientation to a line feature which could be used, for example, to control a mobile robot.

The paper is organized as follows. The remainder of this section discusses prior work in road segmentation and road edge detection. Section II describes the experimental platform for this work and our camera/mirror system. Section III derives the projection of a groundplane line into the image plane, and Section IV describes a gradient based method to adjust line parameters to fit an observation. In Section V we present results from images obtained during experimentation and finally, Section VI concludes.

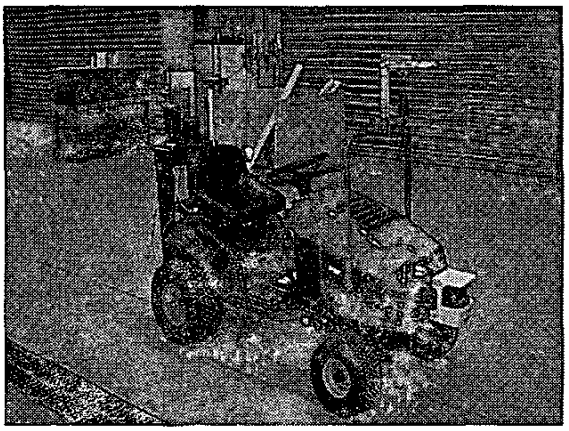

Fig. 1. The experimental platform. Note the omnidirectional camera mounted over the front wheels and the box at the rear which houses the control and computer system.

A. Prior work

Road-following using vision is a relatively well developed field of research with several examples of vehicles capable of traversing thousands of kilometers across the roads of Europe[2] and America [3]. Here we briefly overview some of the image-processing techniques used for road following. For a more complete overview of vision in mobile robots refer to [4], and for road following in general [5].

Early research focused on gradient edge detection to find the road's boundaries, for example in [6]. However, these techniques are very sensitive to noise in the image, caused by, for example, dirt or shadows on the roadway.

Dickmanns made the combination of the 'Gestalt' idea with a 4D-approach, the core of his expectation driven visual servoing approach, and used a geometric model for the road's shape to constrain the search field for the road's edges [7]. Crisman tested a Gaussian colour model for

0-7803-7860-1/03/$17.00 © 2003 IEEE

the road's appearance, combined with a geometric model for roads and intersections [8], which handles shadows very well, and can work on both highways and country roads. Other techniques used in this early study include the multi-resolution approach, nearest mean clustering and a sequential search strategy.

Broggi combined a world-image plane mapping with a template of the road's shape for the purpose of image enhancement, and a directional edge detector with histogramming techniques for the selection of a threshold to segment the road's surface in a colour image [9]. Kaske, Wolf and Husson combined statistical criteria (like energy, contrast, homogeneity and entropy) with a road model (a hyperbolic curve in the image plane), a local extrema search to constrain the search area, and a chi-square fitting of the curve estimation to the observed curve, but admitted that the uncorrelated approach between the two road edges can lead to stability problems [10].

Many of the systems discussed so far use multiple cameras to monitor and react to the vehicle's surroundings. Omnidirectional vision provides a panospheric view of the environment and thus can potentially provide a control system with more complete information, albeit at a reduced resolution when compared to monocular cameras. To date, there are few road (or edge) following systems using panospheric camera's. Gaspar et al, [11] have experimented on a robot operating in corridor environments and demonstrate wall-following behaviours based upon an analysis of the unwrapped panospheric image (in fact they unwrap the image to a birds-eye, plan view of the environment). Das et al. [12] use a technique similar to Horswil's range from height in image technique [13] to reconstruct the orientation and distance to walls in the environment. Chahl and Srinivasan [14] use the ego-motion of the camera to estimate range based on an iterative, optic flow method.

The contribution of our work is to model a line in the environment in the image plane, without unwrapping the panospheric image. We hope to use this in a road/edge following behaviour in our mobile robot.

II. The Platform

The experimental platform is a Toro ride-on mower which has been retro-fitted with actuators, a control system, and a computer, enabling control over the vehicle's operations. All control and computing occurs on-board and the vehicle is fitted with an array of sensors including odometry, differential GPS, a magnetometer, a laser range-finder and an omnidirectional camera; see Figure 1 for a photograph of the vehicle.

A. Camera system

The camera used is from EyeSee 360. The design of this mirror is such that each pixel in the image spans an equal angle irrespective of its distance from the centre of the image — an equiangular mirror. The details of its design are given in [15]. This mirror is slightly different in shape to that of [16], however for the purposes of this experiment, the mirror shape given in [16] is used as it can be described with a closed form solution rather than the numerical form required for the exact shape. For an illustration of the geometry of equiangular mirror optics refer to Figure 2.

The equation describing the surface of such mirrors is [16]:

$$\left(\frac{r}{r_o}\right)^{-\frac{1+\alpha}{2}} = \cos\left[\frac{\theta(1+\alpha)}{2}\right] \tag{1}$$

where the parameters are defined with reference to Figure 2. α is the elevation gain, see (6).

For our system, the mirror was designed to operate with $r_o = 14\,\text{cm}$ with an α value of 11. However, because our mobile robot operates predominately outdoors, we have had to reduce the field of view of the mirror and in the process moved the camera closer to the mirror (reducing r_o) in order to reduce the effect of camera saturation from sunlight. This in effect reduces the angular magnification α and can have an effect on the constancy of α at high angular elevations [16]. However, α remains constant over most of the angular range of the mirror [16].

The panospheric camera model has a number of parameters (r_0, ϕ, α and h) that must be identified. To determine these parameters, we gathered a series of images of a red road cone at specific distances from the camera and determined the corresponding radial distance from the centre of the image (relying on the flat-earth assumption [13])

Using the geometry of the system, we created an estimate of groundplane range given a radial pixel distance. This is given by equation 2:

$$\begin{aligned}\hat{R} &= r_o\left[\cos\left(\tfrac{\theta(1+\alpha)}{2}\right)\right]^{-\frac{2}{1+\alpha}}\sin\theta \\ &+ \left[h + r_o\left[\cos\left(\tfrac{\theta(1+\alpha)}{2}\right)\right]^{-\frac{2}{1+\alpha}}\right]\tan(\alpha\theta+\phi)\end{aligned} \tag{2}$$

where the parameters are defined with reference to Figure 2 and $\hat{R}$ refers to the range estimate (i.e. $d_1 + d_2$). The variable θ is given: by can be found given knowledge of the physical size of the CCD, the outside diameter of the mirror, the size of the image (in pixels), and the radial pixel distance. For our system this relationship is described by:

$$\theta = \arctan\left[\frac{g \times p}{f}\right] \tag{3}$$

where g refers to the radial pixel distance from the centre of the image (measured in pixels) and p refers to the pixel pitch (pixels are assumed to be square).

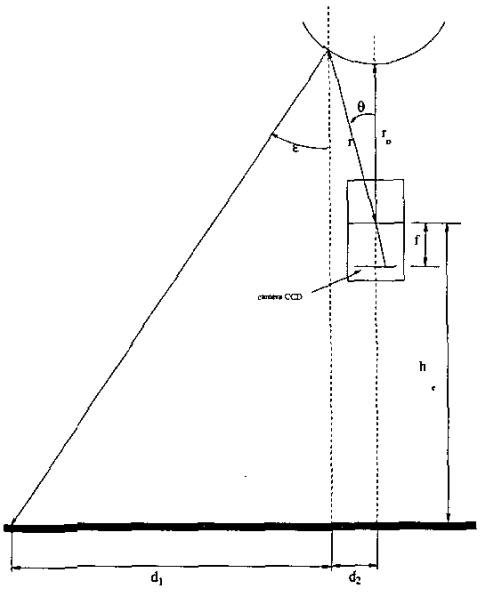

Fig. 2. Geometry of image formation, side view, showing the mirror and perspective camera.

Parameter	Value
r_o	0.0395m
h	1.19m
α	2.74
ϕ	0.0992rad

TABLE I
PARAMETER VALUES AS DETERMINED USING FMINSEARCH IN MATLAB.

By matching the range estimate with the actual measured range, we were able to determine the camera system parameters r_0, ϕ, α and h. Numerical optimisation (Matlab's `fminsearch`) was used to adjust the imaging parameters so that image plane radial distance and measured groundplane radial distance corresponded. The resulting parameters values are shown in Table I.

III. Mapping a groundline to the image plane

We consider a line in the ground plane expressed in parametric form

$$y = mt + c \quad (4)$$
$$x = t \quad (5)$$

with coordinates defined as per Figure 3. The angle between the line and the vehicle's heading direction is ρ where

$$m = \tan\rho$$

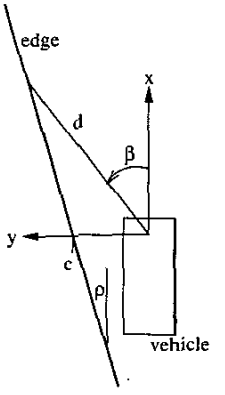

Fig. 3. Plan view of vehicle and line in the groundplane.

The planar distance from any point on the line to our origin, at the camera, is

$$d = \sqrt{x^2 + y^2}$$

where $d = d_1 + d_2$. However, we make the simplifying assumption that $d_2 = 0$.

The elevation angle to the groundline point, from the mirror, is

$$\varepsilon = \tan^{-1}\frac{d}{h}$$

where $h = h_c + r_0$ and again we make a simplifying assumption that the ray from the camera intersects the mirror at a constant height, ie. $r = r_0 \forall\theta$. [1]. The bearing angle to the point is

$$\beta = \tan^{-1}\frac{y}{x}$$

The ray enters the camera at an angle θ from the optical axis which is related to the elevation angle by

$$\varepsilon = \alpha\theta + \phi \quad (6)$$

where α is the elevation gain, and with ϕ is a mirror characteristic.

The radial distance of the point on the image plane is

$$q = f\tan\theta$$

where f is the lens focal length, and the image plane coordinates are

$$u = \frac{q\sin\beta}{p} + u_0$$
$$v = \frac{q\cos\beta}{p} + v_0$$

where p is the pixel pitch and (u_0, v_0) is the image plane coordinate of the principal point. Combining these

[1] In fact it will vary by the height of mirror which is only a couple of cm and is small compared to the mounting height of the camera above the roadway

$$u = f\tan\left(\frac{\arctan(\sqrt{t^2+m^2t^2+2mtc+c^2},h)-\phi}{\alpha}\right)(mt+c)\frac{1}{\sqrt{t^2+m^2t^2+2mtc+c^2}}p^{-1}+u0 \quad (7)$$

$$v = f\tan\left(\frac{\arctan(\sqrt{t^2+m^2t^2+2mtc+c^2},h)-\phi}{\alpha}\right)t\frac{1}{\sqrt{t^2+m^2t^2+2mtc+c^2}}p^{-1}+v0 \quad (8)$$

Fig. 4. Equations of groundplane lines in the image plane.

equations we finally obtain the image plane curve as given by equations (7) and (8).

IV. Fitting the Model to Observation

With respect to the vehicle, we can consider two road edges, each defined in terms of the parameters (m, c). The task is to estimate the values $(\hat{m}, \hat{c})$ given an observed edge, possibly incomplete. An example image, and an estimated line are shown in Figure 5.

We take an initial guess of road edge parameters (m_0, c_0) and adjust them according to the error between observed and estimated image plane curves. To reduce computation we perform a one-dimensional search along horizontal lines looking for the road edge.

The horizontal coordinate of the estimated line is given by

$$u = f(t,m,c)$$

and we can write

$$\Delta u_i = \left.\frac{du}{dm}\right|_i \Delta m + \left.\frac{du}{dc}\right|_i \Delta c$$

which describes the displacement between prediction and measurement in terms of the derivative of the imaging function, which we can derive, and the change in road edge parameters. In order to uniquely identify (m, c) we need at least two equations but in our approach we use a least squares estimate based on the results from many search lines. We can write

$$\begin{bmatrix} \Delta u_1 \\ \Delta u_2 \\ \vdots \\ \Delta u_N \end{bmatrix} = \begin{bmatrix} \left.\frac{du}{dm}\right|_1 & \left.\frac{du}{dc}\right|_1 \\ \left.\frac{du}{dm}\right|_2 & \left.\frac{du}{dc}\right|_2 \\ \vdots & \vdots \\ \left.\frac{du}{dm}\right|_N & \left.\frac{du}{dc}\right|_N \end{bmatrix} \begin{bmatrix} \Delta m \\ \Delta c \end{bmatrix}$$

which we can solve using a matrix pseudo inverse.

We compute Δu_i by the measured displacement between prediction and measurement. This displacement is calculated by searching outwards along each of the horizontal lines for the point with the sharpest drop/rise in intensity, multiplied by a value of 'non-roadness', which is calculated by the pixel's colour using a colour model of the road. Thus, we are looking for edges in intensity that according to their colour don't seem to be part of the road. In practice we would expect these displacements to be noisy and the least squares estimate will deal adequately with this. There are also outlier points due to lighting artifacts, most importantly shadows. We employ a simple filter to eliminate these by rejecting all Δu_i that lie more than 1.5σ from the mean. More complex robust estimation techniques could be applied but this simple rule has been found to work well in practice. We then adjust the roadedge parameters

$$m_{k+1} = m_k + \Delta m$$
$$c_{k+1} = c_k + \Delta c$$

V. Results

Our technique was tested under real-world conditions with images from a white concrete road, surrounded by grass. The road's estimated edges were bootstrapped assuming the camera is located at the centre of the road, looking straight ahead. The estimated edges (shown as lines in the figures), along with the detected edge points (shown as circles) were overlayed on the test images. One intersection and some intense shadows were tested and the road's edges were successfully extracted.

In figure 5, we tested our technique for the most simple case, a straight road with few artifacts. The camera's position is to the left of the road's centre, looking to the left. After 4 iterations for the left edge and 3 for the right one, our technique has converged. In figure 6, the vehicle encounters an intersection coming from the right-hand side. For the left edge, we see how at the section that the road forks, two of the three edge points are dropped as outliers, thus not affecting the correct detection of the edge. Even more impressively, for the right edge, 4 points are dropped as outliers, but it takes 21 iterations to stabilize the edge's location. In figure 7 we try a road scene with partial shadow coverage. Despite a few mis-detections and outliers on both sides , the edges are detected with only minimum error. In figure 8, with much stronger shadows and some bright areas outside the road to the right, we see how the right edge stabilizes off the real edge after 21 iterations, with several mis-detections and outliers. The left edge however is detected correctly and can be used for pose estimation. Our most impressive result appears in image 9, where almost the entire road in our field of view is covered by strong shadows. Both edges are extracted accurately after few iterations, despite some mis-detections.

To sum up, we see that our technique stabilized accurately in almost all of the cases; also, for quite inaccurate

bootstrapping, like in image 5, we see that our technique converged robustly. Partial failures were due to very difficult conditions, particularly in cases with strong, thin shadows, leading to mis-detections that affected the edge's convergence.

However, there are limitations to the detection's robustness. Completely random bootstrapping would not guarantee global convergence; the left edge could be attracted to the footpath to the left, if bootstrapped close enough to it. This problem is solved if we make the assumption that the road is roughly in front of the vehicle, which is pointed approximately in the correct direction. Another limitation is the road's curvature. Our vehicle's maximum speed is less than $3\,\mathrm{ms}^{-1}$, thus it's safe to limit the region of interest to 8 m ahead. In this short road segment, we can ignore the road's curvature and assume it's a straight segment. But for road's with greater curvature, or if we were to increase the roi's outer radius, curvature would have to be taken into account.

VI. Conclusion

This paper has presented a technique for tracking road edges directly in the image plane of an equiangular panospheric camera, that is, without requiring the image to be unwarped. Instead we derived the equation of a curve in the image plane that corresponds to a line in the groundplane. This curve has two parameters which are the angle of the line with respect to the vehicle's heading direction, and its horizontal offset from the line. An iterative method was described to adjust these parameters so as to bring the model curve and the observed curve into alignment. Results using real-world images were presented and show considerable robustness to real-world artifacts such as shadows and intersections.

Acknowledgments

The authors would like to thank the rest of the CSIRO autonomous tractor team: Graeme Winstanley, Leslie Overs, Pavan Sikka, Stuart Wolfe, Stephen Brosnan, and Craig Worthington.

VII. References

[1] K. Usher, P. Corke, and P. Ridley, "Visual servoing of a car-like vehicle – an application of omnidirectional vision," in *Proceedings of the 2002 Australian Conference on Robotics and Automation*, (Auckland, New Zealand), published via CDROM, December 2002.

[2] E. D. Dickmanns, R. Behringer, R. Dickmanns, T. Hildebrant, M. Mauer, F. Thomanek, and J. Shielhlen, "The seeing passenger car 'VaMoRs-P'," in *Proceedings of the International Symposium on Intelligent Vehicles*, (Paris), 1994.

[3] C. E. Thorpe, ed., *Vision and Navigation: The Carnegie Mellon Navlab*. Kluwer Academic Publishers, 1990.

[4] G. N. DeSouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 237–267, February 2002.

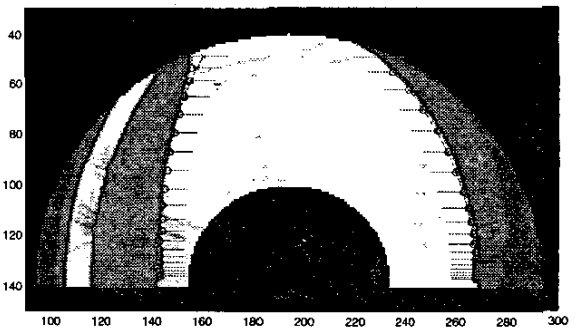

Fig. 5. In a simple example, the edges are detected correctly.

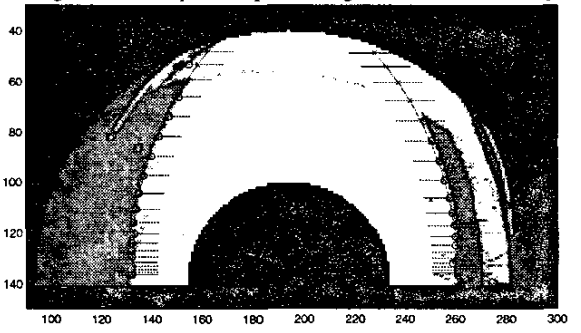

Fig. 6. Coming across an intersection does not adversely affect our results.

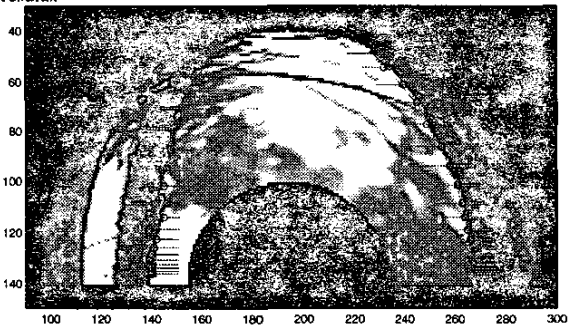

Fig. 7. Our technique isn't affected by shadows.

[5] A. B. M. Bertozzi and A. Fascioli, "Vision-based intelligent vehicles: State of the art and perspectives," *Robotics and Autonomous Systems*, vol. 32, 2000.

[6] S. Liou and R. Jain, "Road following using vanishing points," *CVGIP*, vol. 39, pp. 116–130, 1987.

[7] E. D. Dickmanns, *Active Vision*, ch. 18. Expectation-based Dynamic Scene Understanding, pp. 303–335. The MIT Press, 1993.

[8] J. D. Crisman, *Active Vision*, ch. 7. Color Region Tracking for Vehicle Guidance, pp. 107–120. The MIT Press, 1993.

[9] A. Broggi, "Robust real-time lane detection in critical shadow conditions," in *SCV95*, p. 7A Active Vision I, November 1995.

[10] D. W. A. Kaske and R. Husson, "Lane boundary detection using statistical criteria," *Proceedings of International Conference on Quality by Artificial Vision (QCAV'97)*, 1997.

[11] J. Gaspar, N. Winters, and J. Santos-Victor, "Vision-based

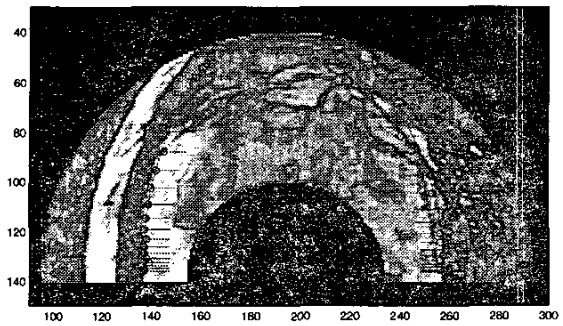

Fig. 8. Thin but strong shadows can lead to a few mis-detections.

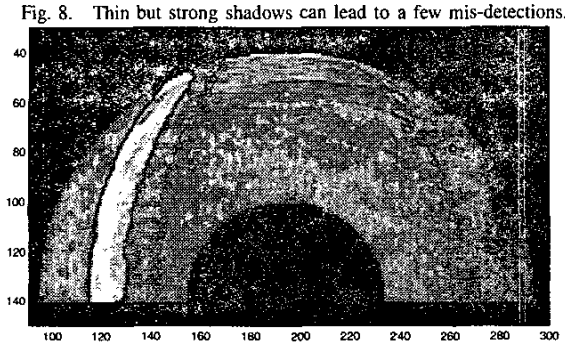

Fig. 9. Even when the roadway is completely covered by shadow, the edges are detected correctly.

navigation and environmental representations with an omnidirectional camera," *IEEE Transactions on Robotics and Automation*, vol. 16, pp. 890–898, December 2000.

[12] A. Das, R. Fierro, V. Kumar, B. Southall, J. Spletzer, and C. Taylor, "Real-time vision based control of a nonholonomic mobile robot," in *International Conference on Robotics and Automation*, (Seoul, Korea), pp. 1714–1719, IEEE, May 2001.

[13] I. Horswill, "Polly: A vision-based artificial agent," in *Proceedings of the eleventh national conference on artificial intelligence (AAAI'93)*, (Washington DC, USA), MIT Press, July 1993.

[14] J. S. Chahl and M. V. Srinivasan, "Range estimation with a panoramic visual sensor," *Journal of the optical society of America*, vol. 14, September 1997.

[15] M. Ollis, H. Herman, and S. Singh, "Analysis and design of panoramic stereo vision using equi-angular pixel cameras," Tech. Rep. CMU-RI-TR-99-04, The Robotics Institute, Carnegie Mellon University, January 1999.

[16] J. S. Chahl and M. V. Srinivasan, "Reflective surfaces for panoramic imaging," *Applied Optics*, vol. 36, pp. 8275–8285, November 1997.