Robot Navigation Using Panoramic Landmark Tracking

Mark Fiala and Anup Basu
Department of Computing Science,
University of Alberta,
Edmonton, Alberta, Canada T6G 2E8
{fiala, anup}@cs.ualberta.ca.
http://www.cs.ualberta.ca/fiala/panotrack/

Abstract

A vision based navigation system is presented for determining a mobile robot's position and orientation using panoramic imagery. Omni-directional sensors are useful in obtaining a 360° *field of view, permitting objects in the vicinity of a robot to be imaged simultaneously. Recognizing landmarks in a panoramic image from an* a priori *model of distinct features in an environment allows a robot's location information to be updated. A system is shown for tracking vertex and line features for omni-directional cameras constructed with* catadioptric *(containing both mirrors and lenses) optics. With the aid of the* Panoramic Hough Transform, *line features can be tracked without restricting the mirror geometry to that which satisfies the single viewpoint criteria. Two paradigms for landmark tracking are explored, with experiments shown with synthetic and real images reported. A working implementation on a mobile robot is shown.*

1 Introduction

One fundamental component for an autonomous mobile robotic platform is to determine its position and orientation with respect to its environment. Example systems use sonar sensors, motor odometry and radio beacons [3]. A passive vision-based system would be very advantageous, and increase the practical utility and scalability of mobile robotics. Hager and Rasmussen define a framework for robot navigation using standard perspective cameras [11, 14]. If this vision system was panoramic, objects all around the robot could be used for finding and updating the position estimate.

The paradigm of an agent translating along a horizontal plane and rotating about a vertical axis, in a space defined and filled with rectilinear polyhedral objects is a reasonable assumption for a mobile robot operating in a man-made environment. Indeed most indoor scenes can be well defined by the primitives of horizontal and vertical lines, corners where such line edges meet, and rectangular surfaces with only horizontal and vertical edges.

Thus we define a framework for creating a passive vision-based navigation system as that of iteratively predicting the location of a prominent geometric feature composed of horizontal and vertical lines, tracking the location of the feature in a panoramic image, and aggregating a number of such observations to arrive at a new position and orientation estimate. We are restricting the problem to that of a mobile agent with motion possible only along the horizontal axis, reducing the dimensions of navigation data to three, two for position and one for orientation. The presence of an *a priori* model of prominent landmarks for predicting trackable features is assumed.

Omni-directional viewing would require many standard narrow field of view cameras, or one panoramic camera. A *catadioptric* optical system is one that uses both lens and mirror components in the optical path, and can be used to capture a 360° field of view around a mobile robot. Basu [2], and others [13, 16, 17, 4, 5] have demonstrated such catadioptric panoramic systems. An example is shown in Fig. 1(left). If a convex, radially symmetric mirror, lens(es) or pinhole and a planar image plane are all mounted along one vertical axis, then a panoramic view can be captured as shown in Fig. 1(right). Such a system has the advantages of processing only one image, and with this image being continuous, not having to deal with discontinuities at the boundaries of views as a ring of conventional cameras would introduce.

Such a view permits the capture of a panoramic view, but with the added challenge of finding and tracking objects in a non-perspective projection. Panoramic catadioptric cameras are not necessarily even cylindrical projections, and locating and tracking features (especially lines) introduces challenges from tracking in more traditional perspective view imagery.

The first positioning method demonstrated assumes a cylindrical projection, or at least a quasi- cylindrical view

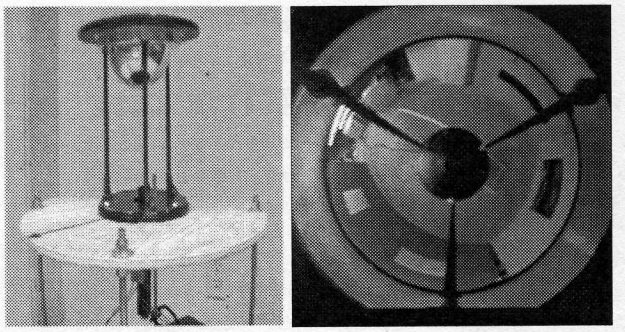

Figure 1: *A Panoramic Imaging system using Catadioptric optics (shown mounted on a mobile platform), and a sample panoramic image.*

and uses vertices as the tracking primitive.

The second method shown is more novel, and it is proposed, a more robust method tracking horizontal and vertical lines, and uses the intersection thereof as the landmark primitive. Issues with using panoramic optics are addressed, specifically that of the absence of the preservation of the straightness of line features, a phenomenon that benefit traditional perspective image analysis. The *Panoramic Hough Transform* is a tool utilized to aid in line detection without restricting the mirror geometry to achieve a pseudo-perspective projection.

2 Triangulating Location From Landmarks

The landmark tracking function provides a set of detected landmarks and their azimuth angle. If the the mobile robot is constrained to movement on a horizontal plane, the elevation of landmarks is used in the tracking, but only the angle is required to find position [6]. Thus the result of landmark tracking need only be a set of landmark labels and their θ_l angles.

If the robot's camera orientation angle θ_c is known, then the position must lie along the line drawn from a detected landmark's world model position, along the azimuth angle $\theta_l + \theta_c$. This line can be described in the $Ax + By + C = 0$ form. The camera location can be determined from the convergence of all such lines. Assuming equal confidence for all θ_l angles, the camera position (X_c, Y_c) can be found by the method of least squares (Eqn. 1), where the quantity minimized is the perpendicular distance from the camera position to all the lines (Eqn. 2).

$$X_c = \frac{-\sum b^2 \sum ac + \sum bc \sum ab}{\sum a^2 \sum b^2 - (\sum ab)^2}$$

$$Y_c = -\frac{\sum a^2 \sum bc - \sum ac \sum ab}{\sum a^2 \sum b^2 - (\sum ab)^2} \quad (1)$$

$$\sum dist^2 = X_c^2 \sum a^2 + 2X_c Y_c \sum ab + \sum c^2 + 2X_c \sum ac + Y_c^2 \sum b^2 + 2Y_c \sum bc \quad (2)$$

The above assumes a known θ_c. It was found to be sufficient to calculate θ_c and (X_c, Y_c) independently since a θ_c value within $+/-45^\circ$ of the correct value yields almost the same position. Newton-raphson iteration with θ_c to find a minimum of Eqn. 2 produces θ_c, which is then used in Eqn. 1 to find (X_c, Y_c).

3 Tracking Vertex Landmarks

In this landmark extraction method, tracking is performed on vertices predicted to be within view. Corners are located by a template matching corner detector function in a warped view of the captured image. A predicted location is given for each corner, plus a descriptor of the expected corner type. Since we are looking for corners of polygons composed of horizontal and vertical edges, there are eight types of corners this descriptor can refer to. The corner can be to the upper left or right, lower left or right, and this corner region can either be lighter or darker compared to its neighborhood.

The useful annular region of source image is warped to a quasi-cylindrical view of dimensions as in Fig. 2a and Fig 3a using bi-linear interpolation. By convention the long rectangular image has pixels representing the azimuth direction in the X direction and radius in the Y direction. This quasi-cylindrical view is a true cylindrical projection only if the mirror profile was parabolic or hyperbolic, because these are the only possibilities for the existence of a virtual perspective point as shown by Baker and Nayar [1]. Note that even in a pure cylindrical projection, the horizontal elements of the corner will not always be horizontal in this image, but it is desired that the corner tracker be as accurate as possible, and that 100 percent performance is not required to still yield an accurate location estimation.

Square n by n sub-images within this warped pseudo-cylindrical view were convolved with an ideal corner of one of these eight types, an operation equal to finding the projection of the sub-image as a vector onto a $n \cdot n$ space as described by Li and Madhavan [15]. This convolution is done over a r_{range} by θ_{range} in X and Y respectively, according to the maximum expected image flow of any landmark feature. These ranges are a function of the distance to, and speed of the mobile robot. The best variance normalized convolution response is chosen as the tracked corner location. It's θ value is chosen and passed onto the triangulation stage.

Optic Flow methods such as finding the minimum of an SSD surface found by correlating image fragments was also attempted, comparing image fragments around the landmark location between frames. However the location of the corner within this window would drift over the duration of the sequence due to the accumulation of matching and round-off error.

The quasi-cylindrical image was created with a width determined from the circumference of the horizon radius rad_h(in pixels). A rough expected error estimate can be made from an assumption of a corner detection accuracy of $Error_{det}$ pixels. With the average distance to a a corner of 150 cm, the camera's position is indeterminate to about $Error_x = 2\pi\frac{Error_{det}}{2\pi rad_h} = \frac{Error_{det}}{rad_h}$ of the average distance to the landmark. If we assume two corner landmarks at 90° angle then the best expected error would be a region of uncertainty with a diagonal width of:

$$Error_{dist} = \sqrt{2}Error_X = \sqrt{2}\frac{Error_{det}}{rad_h} \quad (3)$$

Experiments were performed using synthetic and real image sequences, an example of a section of the quasi-cylindrical view with its tracked corners is shown in Fig. 2a The original image, with the tracked corners annotated is shown for demonstration purposes in Fig. 2b.

The two image sequences and views of the scenes that created them, along with sequences of the quasi-cylindrical warped images and plots of the camera position trajectory can be viewed online at [1].

3.1 Vertex Tracking Results

Experiments were performed with both real and synthetic images.

A synthetically generated sequence of 66 frames, each 500 x 500 pixels, was generated with ray-tracing methods and a simulated catadioptric image sensor. The scene consisted of uniformly shaded rectangular polygons, and an *a priori* model and initial location was provided.

The distance to objects was typically about 70 units, and the error in position had a standard deviation of 0.5 units, a typical error of 0.7 percent. With $rad_h = 157$ pixels, and a corner detection error of $Error_{det} = 1$ pixel, the rough error estimate (Eqn. 3) $Error_x = \frac{Error_{det}}{rad_h} = 0.6$ percent. 3 percent of the location readings were considered outliers with a maximum error of 1.98 units (3.5 percent). Excluding the outliers, the error is as expected. When the diagnostic data was examined, the outliers were found to be a result of failure of the corner detector.

[1] http://www.cs.ualberta.ca/fiala/panotrack/

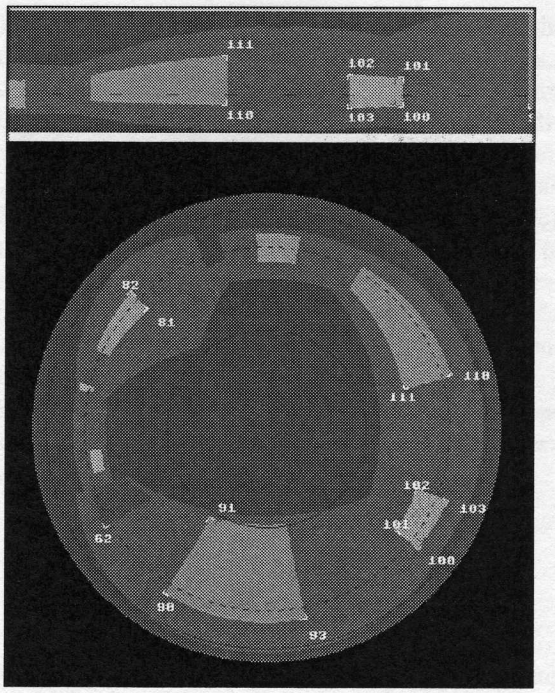

Figure 2: *A: (top) Section of the quasi-cylindrical image created from the panoramic image B: (bottom). The vertices are tracked by a corner detector within a search window from a predicted location.*

The real image experiment consisted of 57 images captured by a panoramic sensor in steps 5 cm apart. The environment was approximately 200 cm x 200 cm, and was composed of distinct light and dark rectangles which were measured. The panoramic catadioptric sensor was composed of a SONY 999 NTSC video camera with a focal length of 590 pixels, and a spherical mirror of radius 4.9 cm, with its center located 15.1 cm from the camera's focal point. Preprocessing was performed on the images (before warping) consisting of averaging with a 3 x 3 filter, and adjusting the aspect ratio. A sample image, with the tracked corners annotated onto the image, and a section of the quasi-cylindrical view, is shown in Fig. 3.

The standard deviation of the error distribution was 5 cm, corresponding to an error of 3 percent over the average distance of 150 cm (from camera to landmark). This would be in accordance with a landmark azimuth error of about $Error_{det} = 5$ pixels, or 0.5 degrees of error. Due to the simple and well spaced corners, only one clearly incorrect outlier position occurred when several of the corner detectors failed at once. The corner detectors often drifted several pixels from the correct corner, and at times settled incorrectly on a corner-like feature. The effects of this were mitigated by the averaging of several landmarks.

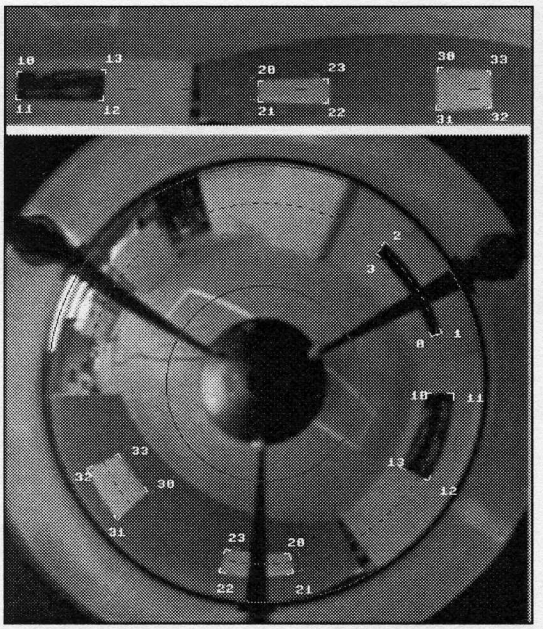

Figure 3: *Real image vertex tracking. A: (top) Section of the quasi-cylindrical image created from the panoramic image B: (bottom).*

A trade off exists between motion accommodated and the possibility of outliers. If the search window is too large, the corner detector may incorrectly locate the corner on another corner-like feature. Likewise if the search window is too small, then the corner will not be located at all. the possibility of finding the wrong corner is greater.

This is a manifestation of the aperture problem encountered in optic flow studies, and it was clear that the method would not be as robust in more cluttered scenes. This motivated the need to examine more of the image in determining landmark location.

4 Tracking Line Junction Landmarks

The previous method served as an introduction to the main method introduced, and shows what can be done without using the *Panoramic Hough Transform*.

Defining a corner landmark as the junction between two successfully tracked line segments decreases the possibility of false landmark detection. Rather than examining a small image segment (30 x 10 pixels in the above experiments) for corner-like properties, tracking the projection of line edges of landmark objects will use image information spanning a much larger area. The probability of a false positive of a corner detection is the much reduced probability of the simultaneous failure of three conditions: two component line edges of the same orientation being both falsely recognized and both meeting at the expected landmark corner position.

Projections of expected line segments forming a corner are predicted according to the current position estimate, and these are then tracked.

However, detecting straight line edges is more challenging in panoramic imagery, especially if the catadioptric sensor does not have a single virtual perspective point. The model of a perspective pinhole projection and the benefits of an affine transform cannot be used in panoramic catadioptric imagery, unless as Baker and Nayar [1] have shown, the mirror has a parabolic or hyperbolic profile. In these two specific cases, light rays are captured whose direction all converge at a virtual perspective point. With all other mirror profiles this is not the case and the direction of captured light rays have no such convergence point, and the optical system is said to be *Non-Single Viewpoint* (non-SVP).

Parabolic and hyperbolic profiles have a single viewpoint and allow the creation of virtual perspective projection views and hence feature extraction and tracking can reduce to methods used in the large body of work directed towards conventional image sensors. An example of what can be done with the geometry of a parabolic profile can be found in [12].

Many other mirror profiles however are desireable for several reasons, an example being circular (spherical mirrors) for the ease of their manufacture. Other useful panoramic mirror profiles are designed to shape the density of image resolution as a function of elevation, either to evenly distribute the image resolution throughout a desired range [7], or to improve resolution at certain elevations. Derrien [8] demonstrated various advantages to relaxing the SVP constraint for panoramic catadioptric system design.

As an alternative to using a mirror profile that guarantees a virtual perspective point, or designing a profile that best approximates one, one can depart entirely from the SVP restriction. Non-SVP profiles could still be used if straight line features can still be recognized, since for the stated problem of mobile robot navigation in man-made environments, they are perhaps the most important feature type. The *Panoramic Hough Transform* models the projection of horizontal straight lines in non-SVP situations, and can be used as a replacement to the pinhole camera paradigm.

Assuming the orientation of the catadioptric sensor having the main axis vertical, horizontal lines can be found with the following theory. Vertical line detection is easier since they project to a straight radial line. Together these two basic primitives, forming most indoor man-made environments, can be tracked to find landmark corners more robustly than corner detection alone.

4.1 Finding Projections of Horizontal Lines in Panoramic Imagery:Panoramic Hough Transform

The Hough transform is a common tool in image processing, its classic application being locating straight lines [18]. In a two-dimensional image, the location of a point alone is insufficient to identify a line, many points are available from an image but it is initially not known which points belong to one of several unknown lines. The classic Hough transform method achieves finding straight lines by recognizing peaks in a parameter space, each point of which representing a possible line in the image. Since a straight line in a two-dimensional image has two degrees of freedom, the set of all possible lines can be represented in a two-dimensional parameter space.

The extension of this concept to locate the projection of straight horizontal lines in the imagery provided by SVP and non-SVP catadioptric panoramic image sensors was presented in [9]. The relative position of a scene point $P(x,y,z)$ to the camera can be expressed as lying along a horizontal line whose closest approach to the camera axis occurs at a direction of θ_{main}. This line is defined by the direction θ_{main}, the distance D_{main} and height Z_{main}. $P(x,y,z)$ can be defined by a fourth parameter $d\theta$ relative to θ_{main}. This is shown in Fig. 4. Note that defining a three-dimensional point with four parameters is not unique and the representation of $P(x,y,z)$ in $\theta_{main}, d\theta, D_{main}, Z_{main}$ has one dimension of freedom representing the family of horizontal lines $P(x,y,z)$ belong to.

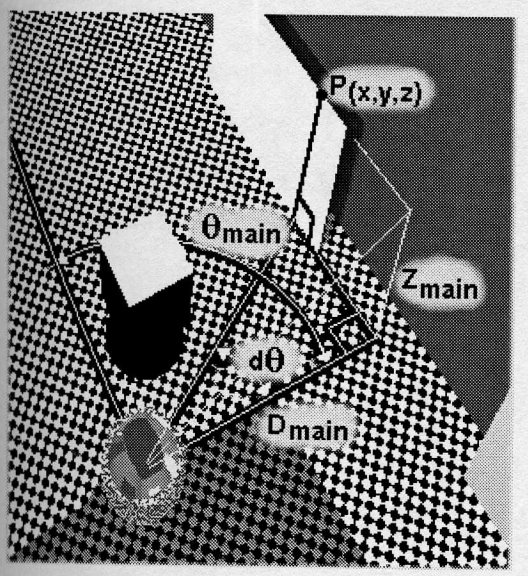

Figure 4: *Basics of Panoramic Hough Transform. Scene point $P(x,y,z)$ can be represented by angle $d\theta$ along a horizontal line defined by $\theta_{main}, D_{main} and Z_{main}$.*

The horizontal line that $P(x,y,z)$ belongs to needs three parameters, θ_{main}, the distance D_{main} and height Z_{main}, which project onto a curved line on the image plane defined by θ_{main} and R_{main}. Due to the loss of depth information, a single parameter R_{main} is a function of D_{main} and height Z_{main} (Eqn. 4). This function depends on the mirror profile, the focal length and the distance from the lens (or pinhole) to the mirror. R_{main} and θ_{main} are enough to define the projection of the horizontal line containing $P(x,y,z)$. Clearly the line itself cannot be defined from this alone, only a plane containing the line can, but since the goal is to recognize, not reconstruct, this is sufficient.

Fig. 5 shows how the scene point $P(x,y,z)$ will appear in the image plane as $P(u,v)$. θ_{main} and $d\theta$ are preserved in the projection.

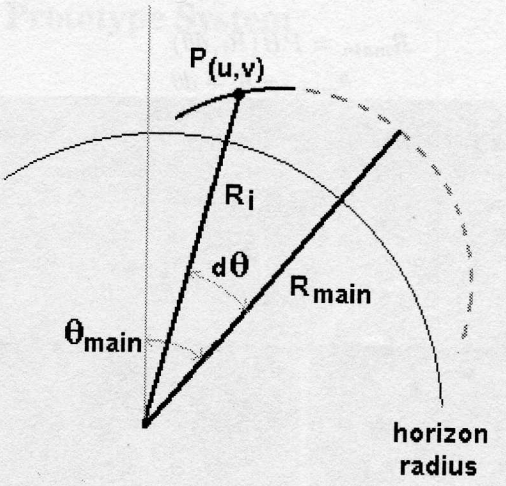

Figure 5: *The projection onto the image plane can be represented by $R_i, \theta_{main} + d\theta$.*

R_{main} and θ_{main} are the X and Y axis of the Panoramic Hough (PH) Transform parameter space. One point in the parameter space corresponds to a the curved family of points in the image space.

Conversely, in the same way that world point $P(x,y,z)$ can be represented by a family of horizontal lines θ_{main}, D_{main} and Z_{main}, its projection on the image plane $P(u,v)$ can be represented by a family of R_{main} and θ_{main} loci. This is similar to the point-line duality in the classic Hough transform, shown graphically in [9].

The additional parameter $d\theta$ is necessary to uniquely define a point along a 3D line, or along the image plane projection. $d\theta$ also uniquely defines a horizontal 3D line from $P(x,y,z)$, and a R_{main} and θ_{main} projection from $P(u,v)$. Due to the radial symmetry, the shape of the projection of a

horizontal line is only a function of R_{main} and $d\theta$, and given in polar coordinates in Eqn. 6.

$$R_{main} = E(\frac{D_{main}}{cos(0)}, Z_{main}) = PH(R_{main}, 0) \quad (4)$$

$$R_{main} = PH(R_i, d\theta) \quad (5)$$

$$R_i = PH^{-1}(R_{main}, d\theta) \quad (6)$$

If $d\theta$ is known, $P(u, v)$ can be expressed in polar coordinates as $P(R_i, \theta_i) = P(R_i, \theta_{main} + d\theta)$, and the parameter point $(R_{main}, \theta_{main})$ is uniquely determined by Eqn. 5. If $d\theta$ is unknown, as it is when a point is considered independently, Eqn. 5 is performed for several $d\theta$ values. This is rewritten in Eqns. 7 and 8, which are used to map a point $P(R_i, \theta_i)$ to all possible $d\theta$ values in the expected range.

$$R_{main} = PH(R_i, d\theta) \quad (7)$$

$$\theta_{main} = \theta_i + d\theta \quad (8)$$

If the PH-Transform is being used for feature extraction within an image [10], $(R_{main}, \theta_{main}$ is calculated and plotted in the transform space for the range $(-\pi/2 < d\theta < \pi/2)$ (Fig. 6). If $\theta_{main})$ is expected to lie within a bounded range $\theta_{min} - \theta_{max}$, $(R_{main}, \theta_{main})$ is only calculated and plotted for points where $\theta_{min} < \theta_i + d\theta < \theta_{max}$.

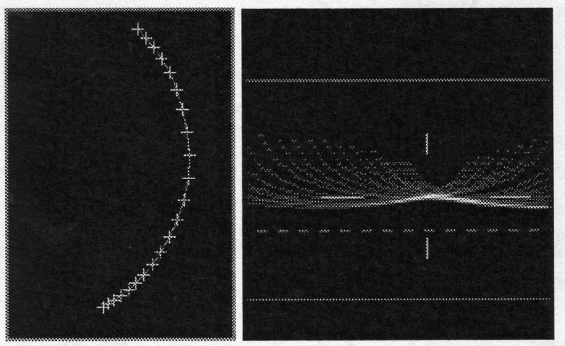

Figure 6: *A (left). A trajectory of points in the source image. Each point maps to a set of points in the PH hough image B (right). If a cluster is found in the PH image (automatically detected peak shown with cross-hairs), then the source image points can lie apon a horizontal line defined by* R_{main} *and* θ_{main}

Source image plane points are shown in Fig. 6 as crosses, and for each image plane point the curved loci of points $(R_{main}, \theta_{main})$ are plotted in the range $\theta_{min} - \theta_{max}$. The magnitude of value added to these mapped points can be unity, or variable according to the confidence of the point (as in tracking), or variable according to edge strength (as in feature extraction). After aggregating the mapping of all source points (Fig. 6a), if a peak is detected (Fig. 6b), then the set of input points (R_i, θ_i) well fit the projection of a horizontal line defined by $(R_{main}, \theta_{main})$.

4.2 Tracking Line Junction Landmarks Using the Panoramic Hough Transform

The above method is applied to landmark tracking as follows. Given a set of polygon corner landmarks deemed visible from the current mobile robot's position, the horizontal and vertical line segments are converted to estimated projection parameters, and these parameters are updated if this line segment is indeed found. A corner landmark is then declared if both segments are found and meet within a threshold error from one another. The horizontal line projection is defined by four parameters: $(R_{main}, \theta_{main}, d\theta_{min}, d\theta_{max})$. The vertical line projection is defined by three parameters $(R_{min}, R_{max}, \theta)$. The radial parameters (R_{main}, R_{min} and R_{max}) are calculated from $R = E(D, Z)$ as in Eqn. 4. An original set of parameters for each segment is predicted, then recognized by the tracking operation, and the parameters are updated if the segment is recognized in the image.

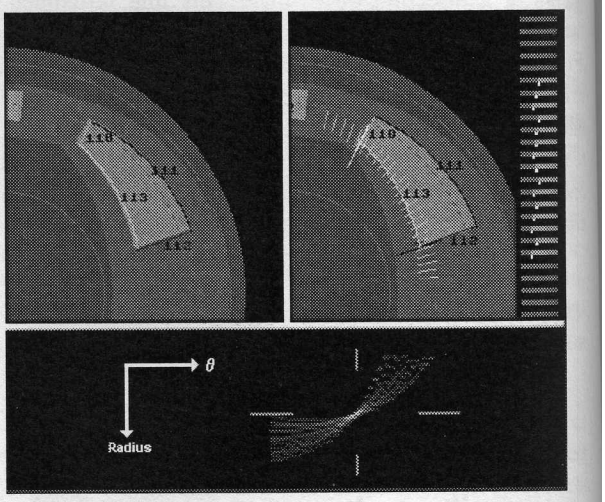

Figure 7: *A (left): The predicted projections of landmark line segments. B (right): The tracking process shown for horizontal line segment projection* 113. *Linear samples are taken (redrawn horizontally to the right). A linear step edge detector reports the presence and location of a step edge of the correct polarity (shown as a white dot beneath its corresponding linear sample). The PH-transform of all these points are plotted in C (bottom), and the new line parameters* R_{main} *and* θ_{main} *found. The new line segment is redrawn in B, and the new endpoints shown as black and white radial lines.*

A set of image plane points are found for each line segment projection. This was accomplished by taking a series

of linear samples at 90° to the predicted line direction. The linear samples are independently analyzed to identify the the possible location of the largest step edge greater than a set threshold. With vertical line projections the samples are taken in the tangential direction, and for horizontal line projections they are taken radially. In both cases, extra samples are taken beyond the predicted end points to accommodate movement.

In the case of vertical lines, the new θ value is calculated by approximating the mode by a truncated median of the angular positions of the step points whose step positions form a contiguous line. For the horizontal line segment, the polar coordinates of the detected step edges are all plotted on the PH-Transform space. If a peak is detected, the line segment's R_{main}, θ_{main} is updated. For both horizontal and vertical line segments the endpoints are updated after the line definition is found, by iteratively taking a linear sample half-way between the last sample that found a step, and the one that did not.

This process is shown below in Fig. 7 for a section of an image from the synthetic image sequence.

The θ_{main} parameter is collected for each successful intersection of a horizontal and vertical line, and the robot's camera position extracted using Eqns 1 and 2.

4.3 Tracking Line Junction Experimental Results

Experiments were performed with the same real and synthetic images sequences as the vertex tracking method.

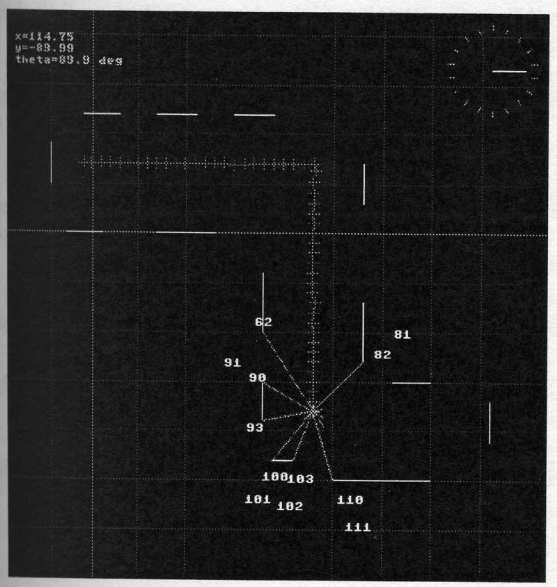

Figure 8: *The recovered trajectory using line junction tracking at frame 44/66 of the synthetic sequence.*

A plot of the recovered trajectory is shown below in Fig. 8. All original and result images can be viewed online. The results can be viewed graphically at [2].

Applying this method to the synthetically generated sequence successfully recovered the camera trajectory, with a standard deviation of 0.4 units, a typical error of 0.5 percent over the average landmark range of 70 units. Repeating the error estimate calculation from Eqn. 3, this indicates just less than sub-pixel accuracy on corner location detection. More important than the improved position accuracy is the absence of outlier points.

The real image experiment using the same image sequence produced an improved standard deviation of distance error of 3.7 cm corresponding to $Error_{det} = 3$ pixels (0.3 degrees of error), and again no outliers.

5 Prototype System

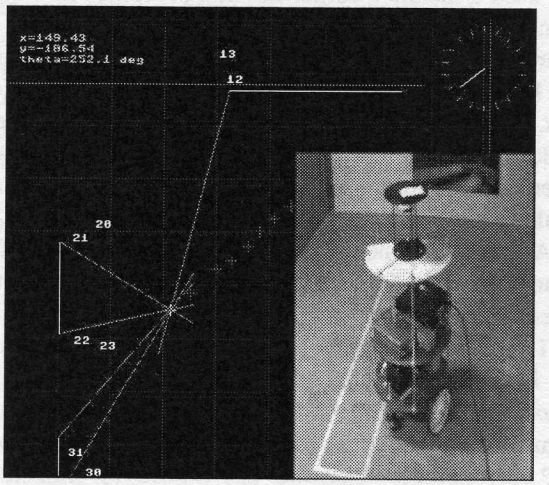

Figure 9: *Screen shot of prototype system showing triangulation of junctions in Fig. 8. The robot's trajectory is drawn by crosses indicating the position at previous frames. In this case the robot is translating towards the lower left.*

A proof-of-concept prototype system using the line junction tracking code was built with a panoramic camera mounted on a mobile robot. A frame rate of about 1.3 hz was achieved, which allowed for reliable tracking with slow movements of the robot. The reliability of the tracking was qualitative, and monitored by watching the estimated position. The robot and a sample screen shot of the robot's extracted position is shown in Fig. 9. The slow frame rate was due mostly to technical issues in the frame grabber and image display, the PH-Transform calculation was done with lookup table and performed rapidly. Two consequences of

[2] http://www.cs.ualberta.ca/fiala/panotrack/

the slow position update rate was that the radial and angular search ranges had to be large, and the robot motion kept slow, especially for rotations. A quick lurch would cause the landmark features to move further than the tracking search range and the position could not be found. With a dedicated real time 30 hz implementation, robust operation of the positioning system is estimated for speeds perhaps greater than 1 metre/second.

6 Conclusions

Two methods of providing position and orientation information for mobile robot navigation with a panoramic camera are presented, and results of a synthetic and real experiment reported. The location detection methods tracked the projections of environment landmarks predicted according to the robot camera's position. The methods both tracked vertices of polygons, but differed in how the vertex was found. The first used a modified corner tracking procedure in a quasi-cylindrical view, and the second tracked line segments and determined corners from the junction of two line segments. The latter was found to be more robust and immune to outliers. Both methods would perform best with as many landmarks as possible. This was found in the vertex tracking experiment where the correct location of some landmarks would reduce the error from false corner measurements.

A prototype mobile robot system was successfully demonstrated having a catadioptric image sensor with a spherical, non-SVP mirror profile.

The novel contribution to mobile robotic navigation lies in the second method where the application of the *Panoramic Hough Transform* allows tracking of straight line segments in catadioptric panoramic cameras free from the SVP mirror profile restriction.

References

[1] S. Baker and S. Nayar. A theory of catadioptric image formation. In *IEEE ICCC Conference*, pages 392–197, 1998.

[2] A. Basu and D. Southwell. Omni-directional sensors for pipe inspection. In *IEEE SMC Conference*, pages 3107–3112, Vancouver, Canada, October 1995.

[3] J. Borenstein, H. Everett, L. Feng, and D. Wehe. Mobile robot positioning: sensors and techniques. 14:231–249, April 1997.

[4] T.J Bruckstein, A.M.; Richardson. Omniview cameras with curved surface mirrors. In *IEEE Proc. Omnidirectional Vision 2000*, pages 79–84, 2000.

[5] M.V Chahl, J.S.; Srinivasan. A complete panoramic vision system, incorporating imaging, ranging, and three dimensional navigation. In *IEEE Proc. Omnidirectional Vision 2000*, pages 104–111, 2000.

[6] D. Cobzas and H. Zhang. 2d robot localization with image-based panoramic models using vertical line features. In *Proc. of Vision Interface*, 2000.

[7] T. Conroy and J. Moore. Resolution invariant surfaces for panoramic vision systems. In *IEEE ICCV Conference*, pages 392–397, 1999.

[8] Steve Derrien and Kurt Konolige. Approximating a single viewpoint in panoramic imaging devices. In *IEEE Proc. Omnidirectional Vision 2000*, pages 85-90, August 2000.

[9] M. Fiala and A. Basu. Hough transform for feature detection in panoramic images. *Technical Report, University of Alberta*, January 2000.

[10] M. Fiala and A. Basu. Line segment extraction in panoramic images. *Technical Report, University of Alberta*, February 2000.

[11] E. Yeh G. Hager, D. Kriegman and C. Rasmussen. Image-based prediction of landmark features for mobile robot navigation. In *IEEE Robotics and Automation*, volume 2, pages 1040–1046, 1997.

[12] C. Geyer and K. Daniilidis. A unifying theory for central panoramic systems. 2000.

[13] M. Yamamoto H. Ishiguro and S. Tsuji. Omniodirectional stereo for making global map. In *3rd Intl. Conf. Computer Vision*, pages 540–547, 1990.

[14] G. Hager and C. Rasmussen. Robot navigation using image sequences. In *AAAI Conference on Artificial Intelligence*, pages 938–943, 1996.

[15] X. Li, C. Shanmugamani, T. Wu, and R. Madhavan. Correlation measures for corner detection. *CVPR*, 86:643–646.

[16] M.-S. Lee M. Nicolescu, G. Medioni. Segmentation, tracking and interpretation using panoramic video. In *IEEE Proc. Omnidirectional Vision 2000*, pages 21-28, 2000.

[17] G. Lacey N. Winters, J. Gaspar and J. Santos-Victor. Omni-directional vision for robot navigation. In *IEEE Proc. Omnidirectional Vision 2000*, pages 21-28, 2000.

[18] V Venkateswar. Extraction of straight lines in aerial images. In *PAMI*, 1992.