Accurate Catadioptric Calibration for Real-time Pose Estimation in Room-size Environments

Daniel G. Aliaga
Lucent Technologies Bell Labs
Murray Hill, NJ 07974
aliaga@bell-labs.com

Abstract

Omnidirectional video cameras are becoming increasingly popular in computer vision. One family of these cameras uses a catadioptric system with a paraboloidal mirror and an orthographic lens to produce an omnidirectional image with a single center-of-projection. In this paper, we develop a novel calibration model that we combine with a beacon-based pose estimation algorithm. Our approach relaxes the assumption of an ideal paraboloidal catadioptric system and achieves an order of magnitude improvement in pose estimation accuracy compared to calibration with an ideal camera model. Our complete standalone system, placed on a radio-controlled motorized cart, moves in a room-size environment, capturing high-resolution frames to disk and recovering camera pose with an average error of 0.56% in a region 15 feet in diameter.

1. Introduction

Recently, we have seen a surge in the use of omnidirectional cameras for applications such as telepresence, 3D reconstruction, and autonomous navigation [3, 4, 17, 18, 20, 21, 22, 23].

Omnidirectional images can be created by several approaches [9, 10, 11, 12, 13]. Nayar [13] proposes a catadioptric system for producing omnidirectional images with a single center-of-projection (COP). In this design, we view a convex paraboloidal mirror (i.e. the parabola's focal point is behind the mirror) with an orthographic projection. The full hemisphere of the field-of-view (FOV), namely 360 by 180 degrees, in front of the mirror is reflected onto an image plane parallel and in front of the mirror. Using simple transformations, we can generate cylindrical and planar re-projections of any part of the FOV, except for an area occluded by the mount near the image center.

In practice, it is difficult to achieve a *perfect* orthographic projection, mainly because of the aperture problem. One approach uses a telecentric lens to create almost ideal orthographic projections but severely limiting the placement and size of the mirror. Another possibility uses a zoom lens combined with magnification lenses to approximate an orthographic projection but this approach is error prone. Unfortunately, both solutions cause substantial errors in pose estimation.

Alternative mirror configurations have also been investigated to ease the orthographic projection requirement. For instance, Bruckstein [5] proposes using two parabolic mirrors. Similar to the single mirror design, the incoming rays bounce off a convex paraboloidal mirror but then are reflected off a second concave paraboloidal mirror centered beneath the first mirror. The reflected rays converge at the second mirror's focal point. A camera, with a standard perspective lens, is placed at the focal point. The system relies on accurate computation of the effective COP for the camera and then on precise localization of the second mirror's focal point. Errors in the localization of the COP cause the camera to capture rays that correspond to slightly non-parallel rays reflected off the first mirror.

Hyperboloid mirrors can also be used [2, 16]. The reflection surface of these mirrors (purposefully) converge rays to a point in front of the mirror. This class of mirrors, which is not so commonly fabricated, also suffers from the same problem -- we must assume to be able to precisely locate the camera at the converging point.

We use an omnidirectional camera based on the design proposed by Nayar [13] and relax the assumption of an ideal projection system for the camera (i.e. perfect orthographic projection and perfect placement) and instead augment the calibration model for paraboloidal catadioptric cameras to compensate for mild perspective projection in addition to radial distortion and mirror misalignment. Consequently, we are able to produce significant improvements in the accuracy of pose estimation.

Our objective is to accurately determine the position and orientation of an omnidirectional camera moving within an average size room, ideally with minimal set up time and

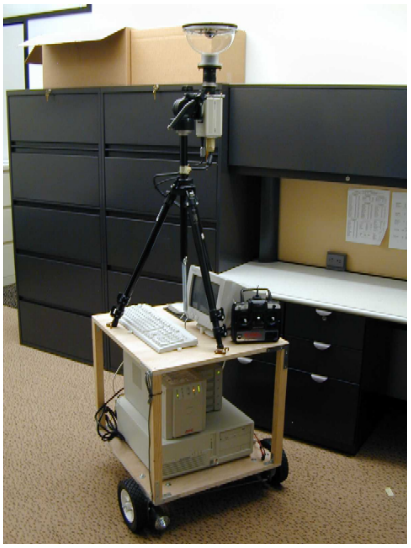

Figure 1. Camera Setup. The paraboloidal mirror is attached to the base of an acrylic dome. The dome is fastened to the video camera. The entire camera is on a radio-controlled motorized cart.

without being too intrusive in the 3D environment. We use our new calibration model together with a robust beacon-based pose estimation algorithm to compute the camera's absolute position and orientation while moving at a fixed eye height, inside a region approximately 5 meters in diameter.

For static scenes, we can loosely group pose estimation algorithms into two categories. The first category uses frame-to-frame changes (e.g., optical flow) to determine the pose offset from one frame to the next. The second category recognizes features of known 3D location and every frame computes pose from the projected position of these features. A large body of literature exists on approaches for standard cameras. Our algorithm falls in the second category and takes advantage of the omnidirectional view of the camera.

Gluckman and Nayar [7], Yagi *et al.* [22], and Kang [8] have proposed alternative catadioptric pose estimation methods. These methods depend on frame-to-frame changes, either optical flow [7, 22] or feature tracking [8], to measure ego-motion. With these approaches it is difficult to obtain robust and accurate absolute pose in room-size environments. Our system (although not self-calibrating, such as in Kang [8]) allows recording multiple discontinuous image sequences and producing consistent absolute pose. Furthermore, our approach extends to multi-room environments, as described in the future work section.

Figure 1 shows a picture of our camera setup. We use a camera based on the commercial S1 design of CycloVision/Remote Reality Incorporated [24]. The camera is placed on a battery-operated motorized cart together with a computer, frame grabber, and fast disk system to store the captured video. The cart is radio-controlled. First, we apply our calibration procedure to the catadioptric system. Then, we place a pair of posts, each equipped with small, bright light bulbs (LEDs are not sufficiently visible at 5 meters) in the corner of the trackable region (e.g., room). Given our calibrated model, we simply need to measure the distance between the posts and then the system triangulates its position and orientation with an average pose error of 0.5%, an order of magnitude improvement over the average pose error of 5% we observed when assuming an ideal catadioptric system.

The rest of the paper is organized as follows. Section 2 and 3 describe our camera model and calibration procedure. Section 4 presents our pose estimation algorithm. Results are listed in Section 5. We compare our results with a theoretical error model in Section 6. Finally, we conclude and outline future work in Section 7.

2. Camera

2.1 Ideal Model

For an ideal imaging system, the camera model is simple. We obtain the mirror center and the mirror radius either manually or by using an automatic algorithm, such as Geyer and Daniilidis [6]. With this information, we quickly determine the distance *d* between the mirror's ideal focal point and the projection of a 3D point *p* onto the XY-plane (i.e., the mirror's reference plane).

Figure 2 illustrates the ideal camera model. A given 3D point $p=(p_x, p_y, p_z)$ reflects off the mirror at $m=(m_x, m_y, m_z)$ and projects onto the image plane at $i=(i_x, i_y)$. Since the mirror is symmetric about the z-axis, we replace m with (m_r, θ, m_z) and p with (p_r, θ, p_z), where $m_r = (m_x^2+m_y^2)^{1/2}$, $p_r = (p_x^2+p_y^2)^{1/2}$, and θ is the angle of the vectors with the x-axis (because of circular symmetry, we can ignore θ for the remainder of this discussion). Since we know the

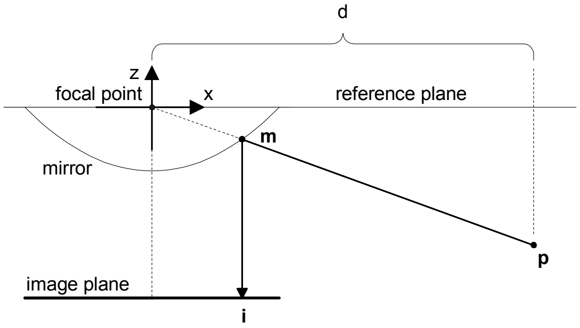

Figure 2. Ideal Camera Model. We depict a paraboloidal catadioptric system and an ideal setup for computing distance between the mirror's focal point and a 3D point.

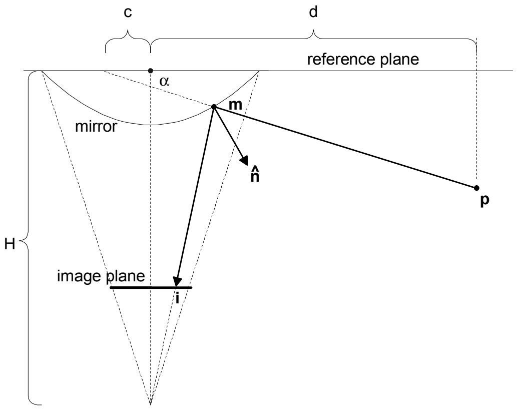

Figure 3. Our Camera Model. We show a paraboloidal catadioptric system that accounts for the (mild amount of) perspective projection that occurs in a practical system. This allows us to more accurately compute the distance between the mirror's ideal focal point and a 3D point.

profile of the mirror, namely the parabola $m_z = m_r^2/(2r) - r/2$ (where r is the mirror radius), we trivially compute (m_r, m_z) from a projected position (i_x, i_y). Then, we use similar triangles to obtain $d = (p_z m_r)/m_z$. With at least two such points, we can triangulate the camera's position. Unfortunately, this scheme does not yield very accurate results. In fact, we observed pose estimation errors close to 45 cms within our trackable region.

2.2 Our Model

Our camera model does not assume a perfect system. Instead, we compute the convergence of the reflected rays (which is equivalent to determining an effective focal length), the distance between the mirror and the image plane, radial distortion, and minor mirror misalignment (e.g., unknown lateral mirror translation and unknown mirror rotation with respect to the mirror reference plane). In addition, because of the mild perspective projection, the mirror reflects objects that are slightly behind the reference plane of the mirror. We provide a formula to compute the subset of the mirror that exactly reflects a hemisphere of the FOV.

Figure 3 depicts our camera model. The points p, m, and i are equivalent to those of Figure 2. In our model, we assume all reflected rays converge at a distance H from the mirror's focal point. Basic optics tells us the incident angle equals the reflected angle. We express this as ($\hat{n}$ is the surface normal at m):

$$\frac{i-m}{\|i-m\|}\cdot\frac{\hat{n}}{\|\hat{n}\|}=\frac{p-m}{\|p-m\|}\cdot\frac{\hat{n}}{\|\hat{n}\|} \qquad (1)$$

For a given 3D point p, a mirror radius r, and a convergence distance H, the free variable in this equation is m_r. We group terms and rewrite this polynomial expression as:

$$\begin{aligned}&m_r^5 - p_r m_r^4 + 2r^2 m_r^3 + (2p_r rH - 2r^2 p_r)m_r^2 + \\ &(r^4 - 4r^2 p_z H)m_r - (r^4 p_r + 2r^3 H p_r) = 0\end{aligned} \qquad (2)$$

We then solve this 5th degree polynomial for m_r [15] and choose the real solution in the range [0, r] and compute m_z using the parabola equation.

With this new information, we more accurately determine the distance d between the mirror's ideal focal point and the XY-plane projection of the 3D point p. We extend the incident ray (i.e. the ray from the 3D point to the mirror) to intersect the mirror reference plane. Because of the perspective projection, the ray overshoots the parabola's focal point by $c=m_z/tan(\alpha)-m_r$, where α is the angle of ray with the XY plane. The final value for d is now:

$$d = (p_z m_r)/m_z - m_z/tan(\alpha) + m_r \qquad (3)$$

The radius r_{180} of the mirror that exactly reflects 180 degrees can be obtained by plugging $p_z=m_z=m_r^2/(2r)-r/2$ into (2). This simplifies the expression to:

$$r_{180} = \sqrt{r(\sqrt{(H^2+4Hr)}-H-r)} \qquad (4)$$

3. Calibration

Our camera calibration procedure consists of three steps: (1) measure numerous 3D points and their corresponding projected positions on the image plane, (2) unreflect the 3D data points using a convergence distance guess so as to make the catadioptric system fit into a pinhole camera model [19], and (3) use an optimization loop [15] to best fit the camera's internal and external parameters to the calibration points. As we change the convergence distance guess during the optimization, we continuously recompute the unreflected 3D position of all the data points. Figure 4 illustrates how we unreflect the points. At the end, the optimized parameters are substituted back into the model.

Our camera model, like Tsai's [19], has eleven variable camera parameters (5 internal and 6 external). We determine the mirror center manually (see Section 5.1). Below we summarize the remaining nine parameters:

- Focal length (f): this is equivalent to the convergence angle of the reflected rays. If the reflected rays were exactly parallel, this value would go to infinity. Also note, that from the final convergence distance, we compute the subset of the mirror that reflects 180 degrees of the FOV.
- Camera-to-world translation (t_x,t_y,t_z): this represents the offset between the image plane and mirror reference plane.
- Camera-to-world rotation (r_x,r_y,r_z): this represents the

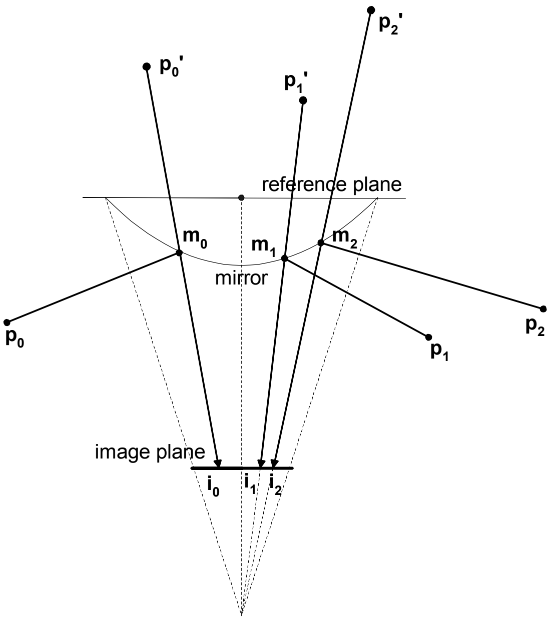

Figure 4. Calibration Procedure. We perform an optimization loop over the camera model's parameters. We fit the calibration points by continuously unreflecting the points using the equations of Section 2 and the current convergence distance guess.

rotation of the mirror reference plane with respect to a assumed world coordinate frame. This does not account for any rotation of the mirror itself with respect to the lens.

- Radial lens distortion coefficient (*k*) and uncertainty scale factor in the horizontal direction (s_x) are carried over from a standard camera.

4. Pose Estimation

Our pose estimation algorithm uses beacons placed in the environment to triangulate camera position and orientation. We place a pair of posts, equipped with small bright light bulbs, in the corner of the trackable region (Figure 5). For our current pose estimation scheme, we only need one bulb per post, but we place two for redundancy. The small light bulbs are easy to track and their projection is at least a few pixels wide at a distance of 15 to 20 feet away. We assume the ground plane is relatively smooth and that the mirror stays approximately parallel to the ground. We arbitrarily place the origin of the world coordinate frame at post 1, at the same height as the mirror reference plane, and align the +x axis with the vector from post 1 to post 2. This approach combined with our calibrated camera model and omnidirectional view computes pose quite accurately.

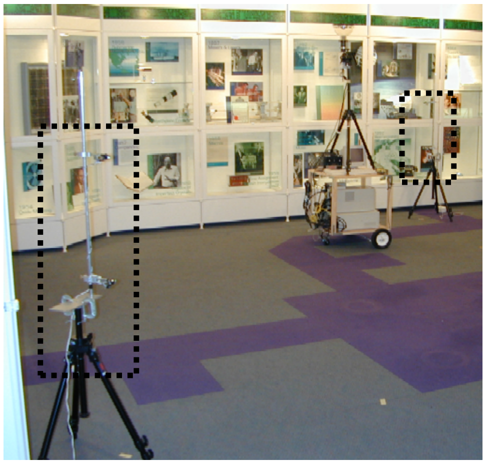

Figure 5. Example Pose Estimation Setup. The cart is placed inside an example environment for calibration and testing. The posts, outlined by boxes in the figure, are placed in the corners of the trackable region.

4.1 Computing Position

We obtain the camera's position by solving an over-determined system. At setup time, we measure the length d_0 of the vector between the two posts. At run time, for each captured frame (see Figure 6), we compute the length of the vectors from the mirror's ideal focal point to the posts (d_1 and d_2) and the angle *A* between the vectors from the focal point to the posts. These three vectors and angle form a triangle. We fit the values into a triangulation and obtain the camera's position.

Our first task is to measure the distance in the XY plane from the focal point to each of the posts. Since we have constructed the motorized cart and the posts, we measure their heights once. For each captured frame, we compute the center-of-mass of each bulb and then input the bulb's image position into the equations of Section 2.2. Afterwards, we use a weighted average of the distances to each bulb. We give more weight to the bottom bulb because an image produced by a paraboloidal mirror more accurately samples the environment towards the middle of the image (bottom of the FOV) than the scenery towards the border of the image (near the horizon of the FOV). Needless to say, the precision ultimately depends on the overall pixel resolution. In Section 6, we discuss system accuracy issues.

Our next task is to compute the XY-plane angle between the posts. To accomplish this, we average for each post the vectors from the mirror center to each of the bulbs. Then, we compute the projected angle *A* between the two vectors.

Our final task is to compute the camera's position. The cosine rule gives us a relationship between the three sides of a triangle (d_0, d_1, d_2) and one angle (*A*) of a triangle. We

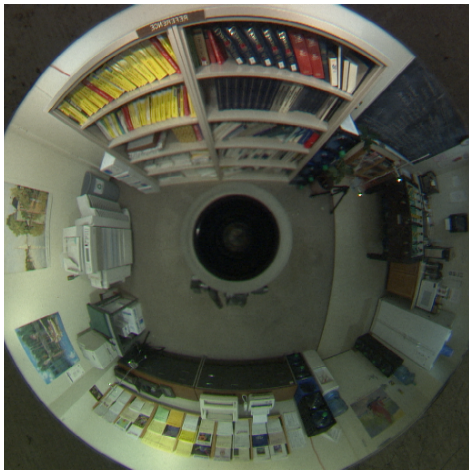

Figure 6. Captured Frame. This is an example frame that could be used for a variety of computer vision algorithms. The bulbs are used to determine the distance to each post and the angle between the vectors from the mirror's center to each post.

assume the ratio of d_1 to d_2 is similar to the ratio of their errors (i.e., shorter distances have smaller errors). Thus, we multiply each distance by a scalar t and substitute the new distance values back into the cosine rule and obtain:

$$t = \sqrt{\frac{d_0^2}{d_1^2 + d_2^2 - 2d_1 d_2 \cos A}} \qquad (5)$$

Now that the three sides and the angle of the triangle agree, we compute camera position (x, y). Figure 7 illustrates the configuration. We also need to determine if the camera is on the positive side (S=1) or the negative side (S=-1) of the x-axis. To this end, we compute in image space the dot product between the normal of the vector to post 1 and the vector to post 2. The sign of the product tells us on which side the camera is located. The following expressions summarize how we compute position given d_0, d_1, d_2, t, and S:

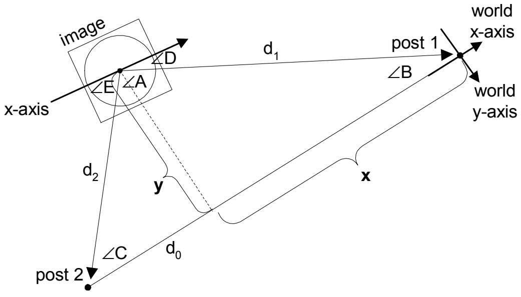

Figure 7. Position and Orientation. (a) To compute position (x,y), we use the (adjusted) distance values d_i and the cosine rule. (b) To compute orientation Ω, we compute the angle between the image's x-axis the world space x-axis.

$$x = \frac{d_0^2 + (td_1)^2 - (td_2)^2}{2d_0}$$
$$y = S\sqrt{(td_1)^2 - x^2} \qquad (6)$$

4.2 Computing Orientation

We compute the camera's orientation in the XY plane by measuring the angle between the world coordinate frame's +x axis and the image plane's +x axis. Figure 7 shows how to compute orientation. Using the distance values d_i, we obtain the additional angles B and C of the triangle. If the image x-axis is aligned with the world x-axis, then angle D equals B and E equals C. In general, D=B+Ω and E=C-Ω, where Ω is the rotation of the image with respect to the world x-axis. For each post, we compute the angles D and E in image space and use a simple averaging of the resulting Ω values to obtain the final camera orientation.

5. Implementation and Results

5.1 Camera

Our panoramic camera uses a high-resolution 3-CCD color video camera (JVC KY-F70, 1360x1024 progressive pixels @ 7.5Hz) and a mirror and lens-system obtained from a commercial S1 unit. We have also used a NTSC-resolution 3-CCD color video camera (Hitachi HV-D25, 640x480 interlaced pixels @ 30Hz). The acrylic dome was machined to ensure the mirror is perpendicular to the camera axis.

To calibrate the camera, we first perform a few empirical measurements. We place the mirror reference plane at a known height and slowly raise a marker placed at a distance until its reflection falls off the edge of the mirror. Then, we measure the marker's height and the distance from the mirror to the marker. This gives us a rough approximation of the convergence of the reflected rays. Next, we measure the diameter and height of the mirror. This allows us to verify that the mirror's focal point is indeed in the middle of the mirror and on the mirror reference plane (height should equal ¼ of the diameter). Using a captured frame, we fit a circle to pass through 20 approximately evenly sampled locations along the mirror's edge and obtain the mirror's center coordinate and radius.

5.2 Calibration

We create calibration data points by moving the cart around a room size environment and capturing frames containing projected points from known 3D locations. We obtain the calibration points by placing the light posts in two corners of the environment and moving the cart to 20 approximately evenly and randomly distributed locations, thus obtaining a total of 80 calibration points. For each

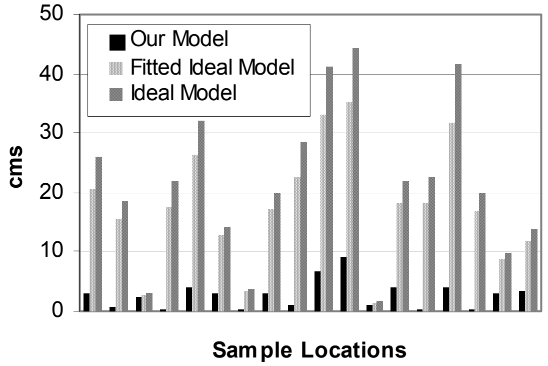

Figure 8. Pose Estimation Error. We show the pose estimation error for three approaches: our (calibrated) model, an ideal model, an ideal model with distance values adjusted in order to fit the pose triangulation together. The sample locations are approximately evenly distributed throughout the trackable region.

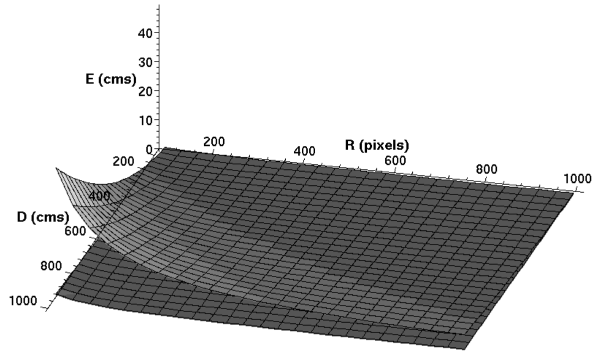

Figure 9. Theoretical Error Model. These surfaces show the limiting accuracy of our pose estimation method, for varying mirror radii R and for varying environment sizes D. The upper surface is due to inaccurate distance to illuminated post estimation and the lower surface is due to inaccurate measurement of the angle between vectors to the posts.

point, we record its projected image position and its 3D world location. We compute the 3D location by measuring the distance between the camera and the posts using plumb lines and tape measures. We estimate our empirical measurements to be accurate to within 1 cm.

We plug all the values into the calibration procedure (Section 3) and run an optimization [15] over the 9 camera model parameters (takes a few seconds on an SGI Indigo2 with a MIPS R4400 @ 250 MHz). We obtain fixed internal camera parameters, such as the CCD chip size and actual number of sensor elements in the CCD array, from the chip specifications. The calibration procedure converges very well for both cameras. By fitting the original calibration points back into the camera model, we observe that the high-resolution camera converges to a maximum deviation of 1.7 mm from any calibration point and the NTSC-resolution camera converges to a maximum deviation of 5 mm. For the high-resolution camera, calibration obtained a convergence distance H=20.67 cms (α=5.59 degrees).

In order to maximize the mirror image-size in the frame, we use a two-pass approach. We run the calibration once, conservatively adjust the mirror and lens to maximize the screen-space used by the subset of the mirror that reflects 180 degrees, and then run the calibration procedure again.

5.3 Pose Estimation

For each tracking session, we place the posts in the corner of the trackable region and measure the distance between the posts. The heights from the ground plane to the mirror reference plane and to each bulb are determined once (both posts are identical) and used for all tracking sessions. We initialize pose by clicking on the bulbs in the first captured frame and subsequently using a simple threshold and center-of-mass computation to track the bulbs. For distances up to 5 meters, the bulbs always appear as high-intensity circular blobs, 5 to 12 pixels wide. Additional lights and specularities can confuse the tracking algorithm but we found the bulbs to work very well in our indoor environments. For the high-resolution camera, we verify position accuracy by empirically measuring the distance between the mirror's center and the posts. Our pose estimation experiments used scale values t ranging approximately from 0.9 to 1.1 (other values, such as c and r_{180}, can be computed from H or α).

We show the results of three pose estimation approaches in Figure 8. First, we show the pose estimation results if we were to assume an ideal catadioptric system (average error is 21.4 cms). Second, we show the results of an ideal system but we fit the computed distance measurements and the measured angle together using equations (5) and (6) (average error is 17.4 cms). Third, we show the pose estimation error of our approach (average error is 2.8 cms). For our test room of 5 meters in diameter, our method has a mean error of 0.56% of the room diameter and a standard deviation of 0.48% of the room diameter.

6. System Accuracy

To better understand the limiting accuracy of our system, we formulate an approximate theoretical error model. In a (catadioptric) pose estimation system the precision ultimately depends on the accuracy of the calibrated camera model and on the resolution of the imaging device. If we assume the calibrated camera model to be fairly accurate, then the relationship between the size of the environment and the resolution of the camera determine the majority of the error. Figure 9 shows approximate error surfaces as a function of mirror radius R (in pixels) and diameter D (in cms) of the trackable region.

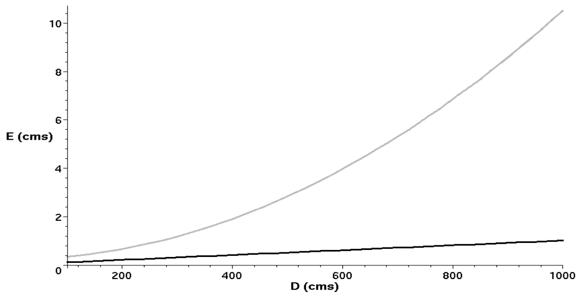

Figure 10. Theoretical Error Model Slice at R=500. This graph is a slice of the error surfaces of Figure 9 for R=500 pixels. For our size environment, D=500 cms, the limiting accuracy is about 3 cms.

The vertical axis represents position error (in cms). The upper surface is the error d_e as a consequence of inaccurate distance estimation to the illuminated posts. The lower surface is the error a_e as a consequence of inaccurate measurement of the angle between the vectors to the posts.

Figure 10 shows the slice of the theoretical error surfaces for a system with *R*=500 pixels, such as ours. Our errors are in the same range as the limiting accuracy. In particular, our average position error of 2.8 cms is about the same as the limiting accuracy of roughly 3 cms (for *D*=500 cms).

How well we can estimate the distance to each post depends on the vertical angular accuracy. In Figure 11(a), we depict the vertical profile of the mirror. If we (conservatively) assume uniform sampling along the image plane and the difference between the camera height and the bulb height is *P* cms then the distance error d_e (in cms) is:

$$d_e = \frac{1}{2}\left(\frac{P}{\tan(\tan^{-1}(\frac{P}{D}) + \frac{\pi}{2R})} - D\right)$$

The precision of the measured angle between the vectors to each post depends on the horizontal angular accuracy. The angular resolution of the system is best at the outer circumference of the mirror (Figure 11b). Thus, we can conservatively measure the distance error a_e (in cms) as a consequence of angular error by:

$$a_e = \frac{1}{2} D \tan(\frac{1}{R})$$

We interpret these distances as the radius of a circle of uncertainty around the mirror's center. As shown in Figure 9, the distance error ranges from under 1 cm of error to over 40 cms of error for a low-resolution image (*R*=100 pixels) in a large trackable region (*D*=1000 cms). For Figures 9 and 10, we assume a reasonable value of *P* equals 150 cms.

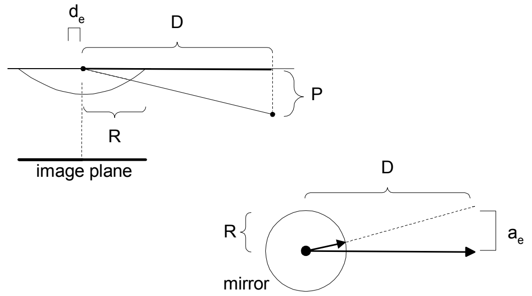

Figure 11. Distance Errors. (a) The upper left diagram depicts how to compute d_e, the limiting positional accuracy as a function to distance to each post. (b) The lower right diagram depicts how to compute a_e, the limiting positional accuracy as a function of horizontal (in the plane of the mirror) angular error.

7. Conclusions and Future Work

We have presented a camera model, calibration procedure, and pose estimation algorithm for a paraboloidal catadioptric system. Our model relaxes the requirement of a perfect orthographic projection, integrates radial distortion correction, and compensates for (minor) mirror misalignment. Our real-time system very accurately recovers the position and orientation of an omnidirectional camera moving in an eye height plane within a room-size environment. The pose estimation method has a brief setup time and does not significantly disturb the surrounding environment. We achieve an order of magnitude improvement over the ideal system. Furthermore, we present a theoretical error model comparing our results to the limiting accuracy of the system.

The captured omnidirectional images are used for a variety of computer vision and computer graphics applications that require accurate camera pose. The degrees of freedom in camera movement is sufficient for a large class of walkthrough-style applications. For instance, our system is currently used as part of a research effort for virtual walkthroughs of existing indoor environments [1].

In the future, we would like to extend the method to include pitch, roll and changing camera height. The two bulbs per post setup is sufficient to detect the presence of pitch, roll, and changing eye height, although the recovered values might not be sufficiently accurate. Robustness and accuracy can be further increased by more posts and bulbs, but at the cost of obstructing the environment to capture. We are also investigating expanding our pose estimation algorithm to multi-room environments by exploiting similarities with the classical art gallery problem [14].

Acknowledgments

I would like to thank the numerous people that have assisted in the construction of the camera and capture system, in particular, Bob Kubli for engineering assistance and Bob Holt for mathematical help. In addition, I thank Ingrid Carlbom for her detailed review of this manuscript and the members of the Visual Communications Department as well as Yuriy Pavlov and George Kiriazides.

References

[1] D. Aliaga, I. Carlbom, "Plenoptic Stitching: A Scalable Method for Reconstructing 3D Interactive Walkthroughs", *Computer Graphics, ACM SIGGRAPH '01*, to appear, August 2001.

[2] S. Baker, S. Nayar, "A Theory of Catadioptric Image Formation", *IEEE International Conference on Computer Vision,* pp. 35-42, 1998.

[3] M. Betke, L. Gurvits, "Mobile Robot Localization Using Landmarks", *IEEE Transactions on Robotics and Automation*, Vol. 13, No. 2, pp. 251-263, 1997.

[4] T. Boult, "Remote Reality Demonstrations", *IEEE Computer Vision and Pattern Recognition*, pp. 966-967, Santa Barbara, CA, June 23-25, 1998.

[5] F. Bruckstein, T. Richardson, "Omniview Cameras with Curved Surface Mirrors", *IEEE Workshop on Omnidirectional Vision*, pp. 79-84, June 2000 (reprint of a AT&T Bell Labs TM of 1996).

[6] C. Geyer, K. Daniliidis, "Catadioptric Camera Calibration", *IEEE International Conference on Computer Vision*, pp. 398-404, 1998.

[7] J. Gluckman, S. Nayar, "Ego-motion and Omnidirectional Cameras", *IEEE International Conference on Computer Vision*, pp. 999-1005, 1998.

[8] S.B. Kang, "Catadioptric Self-Calibration", *IEEE Computer Vision and Pattern Recognition*, pp. 201-207, June 2000.

[9] A. Krishnan, N. Ahuja, "Panoramic Image Acquisition", *IEEE Computer Vision and Pattern Recognition*, pp. 379-384, June 1996.

[10] L. McMillan, G. Bishop, "Plenoptic Modeling: An Image-Based Rendering System", *Computer Graphics, ACM SIGGRAPH '95*, pp. 39-46, August 1995.

[11] K. Miyamoto, "Fish Eye Lens", *Journal of Optical Society of America*, 54(8):1060-1061, August, 1964.

[12] V. Nalwa, "A True Omnidirectional Viewer", Technical Report, Bell Labs, Holmdel NJ 07733, February 1996.

[13] S. Nayar, "Catadioptric Omnidirectional Camera", Proc. of *IEEE Computer Vision and Pattern Recognition*, pp. 482-488, 1997.

[14] J. O'Rourke, **Art Gallery Theorems and Algorithms**, Oxford University Press, New York, 1987.

[15] W. Press, S. Teukolsky, W. Vetterling, B. Flannery, **Numerical Recipes in C**, Cambridge University Press, 2nd edition, 1999.

[16] D.W. Rees, "Panoramic Television Viewing System", United States Patent No. 3,505,465, April 1970.

[17] J. Shimamura, N. Yokoya, H. Takemura and K. Yamazawa, "Construction of an Immersive Mixed Environment using an Omnidirectional Stereo Image Sensor", *IEEE Workshop on Omnidirectional Vision*, pp. 62-69, June 2000.

[18] C. Taylor, "VideoPlus", *IEEE Workshop on Omnidirectional Vision*, pp. 3-10, June 2000.

[19] R. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses", *IEEE Journal on Robotics and Automation*, RA-3:4, pp. 323-344, 1987.

[20] N. Winters, J. Gaspar, G. Lacey and J. Santos-Victor, "Omni-Directional Vision for Robot Navigation", *IEEE Workshop on Omnidirectional Vision*, pp. 21-28, June 2000.

[21] Y. Yagi, S. Kawato, and S. Tsuji. "Real-time Omnidirectional Image Sensor (Copis) for Vision-guided Navigation", Trans. *On Robotics and Automation*, 10:11-22, 1994.

[22] Y. Yagi, w. Nishii, K. Yamazawa, and M. Yachida, "Rolling Motion Estimation for Mobile Robot by Using Omnidirectional Image Sensor HyperOmniVision", *Int'l Conference on Pattern Recognition*, pp. 946-950, 1996.

[23] Y. Yagi, Y. Nishizawa, M. Yachida, "Map-Based Navigation for a Mobile Robot with Omnidirectional Image Sensor COPIS", *IEEE Transactions on Robotics and Automation*, Vol. 11, No. 5, pp. 634-648, 1995.

[24] Remote Reality Inc. (formerly CycloVision), http://www.remotereality.com, Westborough MA.