Non-Metric Calibration of Wide-Angle Lenses and Polycameras *

Rahul Swaminathan and Shree K. Nayar

Department of Computer Science, Columbia University
New York, New York 10027
Email: srahul@cs.columbia.edu, nayar@cs.columbia.edu

Abstract

Images taken with wide-angle cameras tend to have severe distortions which pull points towards the optical center. This paper proposes a method for recovering the distortion parameters without the use of any calibration objects. The distortions cause straight lines in the scene to appear as curves in the image. Our algorithm seeks to find the distortion parameters that would map the image curves to straight lines. The user selects a small set of points along the image curves. Recovery of the parameters is formulated as the minimization of an objective function which is designed to explicitly account for noise in the selected image points. Experimental results are presented for synthetic data with different noise levels as well as for real images. Once calibrated, the image streams from these cameras can be undistorted in real time using look up tables. We also present an application of this calibration method for wide-angle camera clusters, which we call polycameras. We apply our distortion correction technique to a polycamera with four wide-angle cameras to create a high resolution 360 degree panorama in real-time.

1 Introduction

In many vision applications, such as surveillance, it is desirable to capture the entire region of interest with as few cameras as possible. Wide-angle cameras help in this regard, but at the cost of severe image distortions. Wide-angle lenses that adhere to perspective projection would necessitate the use of prohibitively large image detectors. To work around this problem, wide-angle lenses are designed to severely bend rays of light around the periphery of the field of view[1], thus permitting the use of a small image detector (say, a CCD). The effects of the resulting image distortions are clearly visible in Figure 1.

If the optics of the wide-angle camera system are known

---

*This work was supported in parts by the VSAM effort of DARPA's Image Understanding Program and an ONR/DARPA MURI grant under ONR contract No. N00014-97-1-0553.

[1] Severe bending of light rays typically leads to a non-singular entrance pupil. The resulting locus of pupils in three dimensions is called a *diacaustic* [Born and Wolf, 1965]. This implies that, for a wide-angle lens, complete removal of distortions cannot be achieved. For our purposes, we will assume a small pupil locus that can be approximated by a single point.

Figure 1: Images captured with wide-angle cameras have severe distortions that alter the appearance of objects in the scene.

apriori (i.e. the distortion parameters), then distortion correction can be easily applied. Unfortunately, such information is seldom revealed by manufacturers. Furthermore, in mass production, optical characteristics are sure to vary from one lens to the next. It is therefore desirable to have a simple calibration method for extracting the distortion parameters. This paper presents such a calibration method.

Many calibration methods have been suggested for recovering lens distortion parameters. Tsai [1987] used known points in 3D space to recover some of the distortion parameters. Goshtasby [1989] utilized Bezier patches to model the distortions and used a uniform grid placed in front of the camera as a calibration object. Weng [1992] also used calibration objects to extract all the distortion parameters. All these methods fall in the category of "stellar" calibration, where objects with points of known relative coordinates are used.

In contrast, Brown [1971] proposed a "non-metric" approach that does not rely on known scene points. Instead, he relies on the fact that straight lines in the scene must always perspectively project to straight lines in the image. An iterative least-squares formulation is used to estimate distortion parameters which map distorted image curves to straight lines. Brown's algorithm relies on essentially noiseless image data, which is obtained by imaging plumb-lines suspended against a black background onto a photographic plate. More recently, Kang [1997] used snakes to represent the distorted curves instead of discrete points.

413

Becker [1995] used three mutually orthogonal sets of parallel lines and a vanishing point constraint to recover distortion parameters. Stein (see [1995] [1997]) used , point correspondences in multiple images are used to estimate radial distortions. A novel approach to estimating and correcting for distortion parameters is that proposed by Sawhney in [1999]. He uses more direct techniques without relying on point correspondences or features.

Previous work suffers from one or more of the following restrictions: calibration objects need to be used, not all the distortion parameters are recovered, or the algorithm is highly sensitive to noise. One exception is the work of Becker [1995]. However, Becker's constraint (triplets of orthogonal lines) is less abundant in urban settings than the randomly oriented straight lines we use. In addition, while Becker uses the Normal distribution to model noise, our algorithm makes no assumptions about the exact nature of noise in the selected image points. We formulate the estimation of distortion parameters as the minimization of a noise insensitive objective function via efficient search.

Experimental results with synthetic and real data are presented, which demonstrate the robustness of the proposed method in the presence of large amounts of noise. In addition, we describe a useful application of our calibration technique. We present the notion of a polycamera, which uses a tight cluster of multiple cameras to capture a large fully connected field of view. Wide-angle cameras are useful in this context as they minimize the number of cameras needed to cover the desired field of view. We describe a panoramic polycamera constructed using four wide-angle cameras. First the distortion parameters of each camera are recovered. Then these parameters are used to map the outputs of the four cameras to a single high-resolution panorama, in real time.

2 Distortion Model

Let the perspective projection of a scene point be q' (see Figure 2). Due to distortions in the lens, the point gets mapped to the point q. Let (x, y) be the Cartesian and (r, ϕ) be the polar coordinates of q. Similarly, let (x', y') be the Cartesian and (r', ϕ') be the polar coordinates of q'. Also, let the optical center C be located at (x_p, y_p). Then, the Cartesian and polar coordinates are related as:

$$r = \sqrt{\bar{x}^2 + \bar{y}^2} \ , \ \tan(\phi) = \frac{\bar{y}}{\bar{x}} \ ,$$

where:

$$\bar{x} = x - x_p \ , \ \bar{y} = y - y_p \ . \tag{1}$$

The distortion of q' can be split into three components: (a) shift of the image center, (b) radial distortion, and (c) decentering distortion. The first of these is given by (x_p, y_p), while the remaining two we will now describe in greater detail.
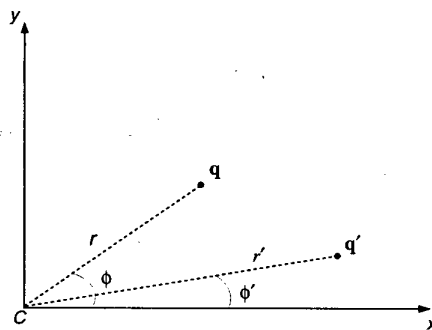

Figure 2: q' is the perspective projection of a scene point onto the image plane. Due to radial and decentering distortions, q' gets mapped to the point q.

2.1 Radial Distortions

Radial distortions in most wide-angle cameras pull points towards the optical center. This kind of distortion is also referred to as *barrel distortion* [Born and Wolf, 1965]. This effect is radially symmetric and depends solely on the distance from the optical center. The radial distortion present in the point q can be written as:

$$\Delta r(q) = \sum_{i=1}^{\infty} C_{2i+1} r^{2i+1} \ , \tag{2}$$

where, C_{2i+1} are the distortion parameters. Terms higher than the fifth-order one can be ignored as their contribution to the distortion is negligible in practice [Brown, 1966]. Hence, we have:

$$\Delta r(q) \approx C_3 r^3 + C_5 r^5 \ . \tag{3}$$

2.2 Decentering Distortions

Decentering distortions are caused by the non-orthogonality of the lens components with respect to the optical axis. It is highly unlikely for an imaging system to have no decentering distortions, which, unlike radial distortions, act tangentially. We use Conrady's model [Conrady, 1919] for decentering distortion:

$$\Delta T_x(q) = [P_1 r^2 (1 + 2\cos^2(\phi)) + 2P_2 r^2 \sin(\phi)\cos(\phi)]$$
$$[1 + \sum_{i=1}^{\infty} P_{i+2} r^{2i}]$$
$$\Delta T_y(q) = [P_2 r^2 (1 + 2\sin^2(\phi)) + 2P_1 r^2 \sin(\phi)\cos(\phi)]$$
$$[1 + \sum_{i=1}^{\infty} P_{i+2} r^{2i}] \ , \tag{4}$$

where, P_i are the distortion parameters and ΔT_x, ΔT_y are the distortions along the x and y directions, respectively.

The higher-order terms in the above expression are relatively insignificant. Hence, P_1 and P_2 are generally sufficient for modeling decentering [Brown, 1966]:

414

$$\Delta T_x(\mathbf{q}) \approx [P_1 r^2(1 + 2\cos^2(\phi)) + 2P_2 r^2 \sin(\phi)\cos(\phi)]$$
$$\Delta T_y(\mathbf{q}) \approx [P_2 r^2(1 + 2\sin^2(\phi)) + 2P_1 r^2 \sin(\phi)\cos(\phi)] . \quad (5)$$

2.3 Complete Distortion Model

The total distortion is the sum of the above components:
$$\Delta x(\mathbf{q}) \approx \cos(\phi)[\Delta r(\mathbf{q})] + \Delta T_x(\mathbf{q})$$
$$\Delta y(\mathbf{q}) \approx \sin(\phi)[\Delta r(\mathbf{q})] + \Delta T_y(\mathbf{q}) . \quad (6)$$

In order to correct distortions, we need to recover the parameters $(C_3, C_5, P_1, P_2, x_p, y_p)$.

3 Objective Function Formulation

The constraint used in this paper is that, under perspective projection, straight lines in the scene should project to straight lines in the image. We assume that the user of our calibration method knows which (distorted) image curves correspond to straight lines in the scene. Based on this knowledge, the user selects points along these curves. In this setting, an objective function can be defined which, when minimized, yields the parameters that undistort the curve points to lie on straight lines.

We present three objective functions, namely, sum of squared distances (from straight lines), normalized sum of squared distances and one that explicitly estimates noise in the chosen image points. The first two are presented mainly to demonstrate that simple objective functions (similar to ones proposed previously) are highly noise sensitive. In contrast, the third function is designed to explicitly account for noise in the image points chosen by the user. All our objective functions are non-linear and are minimized using efficient search algorithms. In what follows, our goal will be to recover only the radial and decentering distortion parameters. The shift of the optical center (x_p, y_p) will be recovered separately in an iterative fashion.

3.1 Sum of Squared Distances (ξ_1)

This objective function is similar to the one used in the iterative least-squares method developed by Brown [1971]. In our approach, during search, a set of (hypothesized) distortion parameters $\mathcal{S} = \{C_3, C_5, P_1, P_2\}$ are applied to the selected image points $\mathbf{q}(x, y)$. Lines are fitted to the resulting points $\mathbf{q}'(x', y')$ and the objective function is computed as the sum of the squared distances of the points from their corresponding "best-fit" lines.

Let the best-fit line for a set of points $\mathbf{q}'$ (originally, on the same image curve) be parameterized by (θ, ρ), where θ is the angle the line makes with the horizontal axis and ρ is the distance of the line from the image center. Therefore, the error due to a single undistorted point $\mathbf{q}$ is given by:
$$e = (x' \sin(\theta) - y' \cos(\theta) + \rho)^2 ,$$
where:

$$x' = x + \Delta x(\mathbf{q}), \quad y' = y + \Delta y(\mathbf{q}) . \quad (7)$$

Let the number of curves selected by the user be L, and the number of points on each line l be P_l. Then, the objective function is given by:
$$\xi_1 = \sum_{l=1}^{L} \sum_{p=1}^{P_l} \left(x'_{p,l} \sin(\theta_l) - y'_{p,l} \cos(\theta_l) + \rho_l \right)^2 . \quad (8)$$

where θ_l and ρ_l are the best-fit line parameters corresponding to image curve l and $(x_{p,l}, y_{p,l})$ is the pth point on line l.

3.2 Normalized Sum of Squares (ξ_2)

Although simple, the above formulation is very sensitive to noise. From the distortion model, it can be seen that noise is magnified by the higher-order distortion terms in $\mathcal{S}$ (in particular, the third-order and fifth-order terms). As a result, points that lie closer to the image center contribute less to the error than points farther away. This effect is partially remedied by normalizing the error e in (7) by the square of the distance ρ_l of the corresponding line l from the image center. Then, the modified objective function is:
$$\xi_2 = \sum_{l=1}^{L} \sum_{p=1}^{P_l} \left(\frac{x'_{p,l} \sin(\theta_l) - y'_{p,l} \cos(\theta_l) + \rho_l}{\rho_l} \right)^2 . \quad (9)$$

3.3 Explicit Noise Estimation (ξ_3)

The objective functions ξ_1 and ξ_2 are defined in the space of the undistorted points (i.e. after applying $\mathcal{S}$). Since noise in the selection process is induced in the distorted coordinates, it is more appropriate to formulate an objective function that uses errors computed in the space of distorted image points, so as to avoid non-linear biases inherent to our distortion model.

As shown in Figure 3, let $\mathbf{q}$ be the distorted point under consideration and $\mathbf{q}'$ be the "undistorted" point obtained by applying the set of distortion parameters $\mathcal{S}$. Again, l is the best-fit line for all the points $\mathbf{q}'$, which are believed to lie on the same scene line. We now determine (via search) the point $\hat{\mathbf{q}}$ close to $\mathbf{q}$, which when undistorted using $\mathcal{S}$ would lie on l (at $\hat{\mathbf{q}}'$ in Figure 3). The new error function is defined as:
$$e = \|\mathbf{q} - \hat{\mathbf{q}}\|^2 . \quad (10)$$

Since $\hat{\mathbf{q}}'(\hat{x}', \hat{y}')$ must lie on l, it must satisfy the constraint:
$$\hat{x}' \sin(\theta) - \hat{y}' \cos(\theta) + \rho = 0 ,$$
where:
$$\hat{x}' = \hat{x} + \Delta x(\hat{\mathbf{q}}), \quad \hat{y}' = \hat{y} + \Delta y(\hat{\mathbf{q}}) . \quad (11)$$

Using all the selected points, the objective function is determined as:
$$\xi_3 = \sum_{l=1}^{L} \sum_{p=1}^{P_l} \|\mathbf{q}_{p,l} - \hat{\mathbf{q}}_{p,l}\|^2 . \quad (12)$$
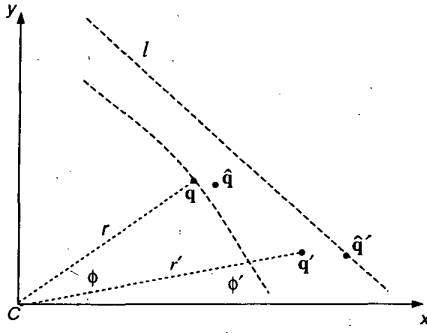
Figure 3: q is a point selected by the user and q' is its undistorted location on applying the (hypothesized) distortion parameters S. l is the "best-fit" line estimated for all q', which are believed to lie on the same scene line. $\hat{q}$ is a point close to q such that, its undistorted location $\hat{q}'$ (obtained by applying S on $\hat{q}$) lies on l. We wish to minimize the distance between q and $\hat{q}$.

4   Minimization of ξ_1, ξ_2 and ξ_3

We now describe the non-linear search algorithms used to recover the distortion parameters S by minimizing the objective functions ξ_1 , ξ_2 and ξ_3. It should be noted that our calibration method is in no way restricted to the specific search algorithms we have used.

We used a modified simplex search algorithm outlined in [Nelder and Mead, 1965], implemented in the IMSL library. This implementation requires the user to provide upper and lower bounds on the parameters to be estimated. The following bounds were used: $C_3 : (-10^{-5}, 10^{-5})$, $C_5 : (-10^{-9}, 10^{-9})$, $P_1 : (-10^{-5}, 10^{-5})$, $P_2 : (-10^{-5}, 10^{-5})$. These bounds are highly conservative as they include distortions that are significantly more severe than those found in typical wide-angle imaging systems.

At each step of the non-linear search, given the set of (hypothesized) parameters S, we must compute the objective function. Computation of ξ_1 and ξ_2 is straightforward, using a linear least-squares method to fit the lines l. However, computing ξ_3 also requires the estimation of the points $\hat{q}$ (see (12)), for which there is no closed-form solution.

To this end, we solve for each $\hat{q}_{p,l}$ by searching the neighborhood of $q_{p,l}$ for the point which, when undistorted using S, lies on l. This requires a $2D$ search (see Figure 3), which is computationally intensive. For efficiency, we use a $1D$ search, since there always exists a point in the radial direction of the selected point q, which lies on the true distorted curve. This approximation enables us to estimate the distortion parameters (C_3, C_5, P_1, P_2), in under 30 seconds on a 300MHz PC.

Note that we did not include the optical center $C(x_p, y_p)$,

in the non-linear search for the distortion parameters S. Initial experiments revealed that including $C(x_p, y_p)$ can produce unstable results in the presence of noise, due to the higher dimensionality of the search space. Therefore, we have nested the estimation of (C_3, C_5, P_1, P_2) within a coarse-to-fine search for the optical center (x_p, y_p).

5   Experimental Results : Synthetic and Real Images

To evaluate the robustness of our calibration technique, it is imperative to test it in the presence of noise. Noise enters the system from three main sources: human error in selecting points in the image, finite image resolution, and the fact that lines in the scene may not be perfectly straight. It is difficult to quantify the robustness of any non-metric calibration method using only real images, due to lack of ground truth. Hence, we synthetically generate the noisy image points that are used as inputs to our algorithm.

Points were randomly sampled from synthetically generated lines with random orientations and positions (see Figure 4(a)). Using known distortion parameters, the sampled points were distorted (see Figure 4(b)). To simulate erroneous point selection, we added uniform noise in the interval $(-w, +w)$ to the coordinates of the points (see figure 4(c)). We then used our algorithm to estimate the distortion parameters from the noisy data and used these parameters to undistort the noiseless image points (see Figure 4(d)).

Although precise recovery of the distortion parameters S ensures an exact match between the sampled points (Figure 4 (a)) and the undistorted points (Figure 4 (d)), it is not necessary for accurate distortion correction. A good measure of accuracy and robustness is the distance between the initial position (see Figure 4(a)) and the recovered position (see Figure 4(d)) for each point. We tested each objective function using lines $\mathcal{L}$ of different orientations and positions , various distortion parameters S and several noise levels in the interval $w = [0, 5]$ pixels.

Tables 1 (a)-(d) show the errors present in the recovered undistorted points using the sum of squares (ξ_1), normalized sum of squares (ξ_2) and the noise estimation method (ξ_3), respectively. Errors are defined as the average of the absolute distances between each of the undistorted points and the original sampled points. Notice the sharp degradation in accuracy with increasing noise in the simple sum of squares approach (ξ_1) (see Table 1(a)). Although ξ_2 performs better than ξ_1 for certain noise levels, it breaks down for high levels of noise. In contrast ξ_3, is much more robust and yields sub-pixel accuracy even for high noise levels, as can be seen from Table 1(c). More results obtained by using ξ_3 are shown in Table 1(d).

As mentioned earlier, recovery of the optical center is implemented as a coarse-to-fine exhaustive search around the
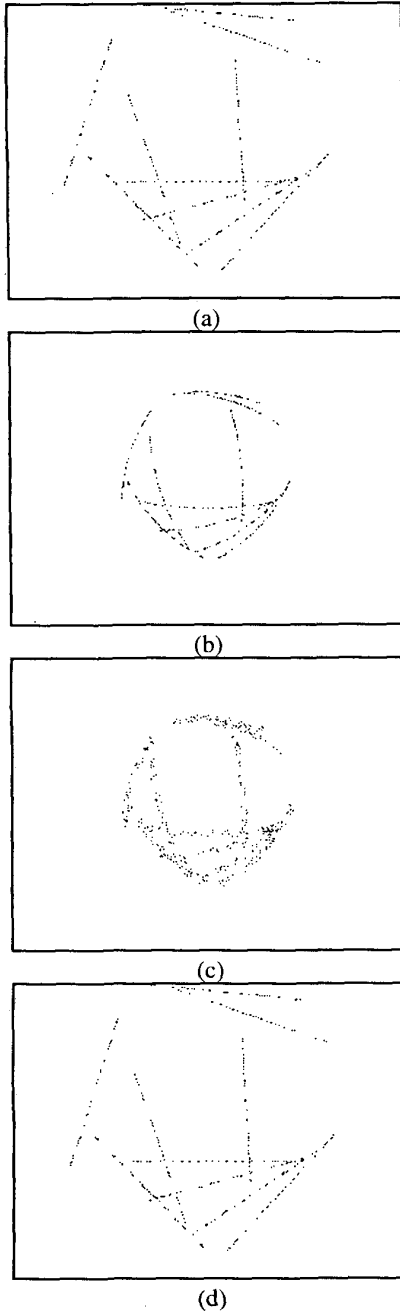
416

(a)

(b)

(c)

(d)

Figure 4: (a) Points randomly sampled from synthetically generated lines. (b) Known distortions are applied to the points in (a). (c) Uniformly distributed random noise in the interval (-5 pixels, 5 pixels) is added to the distorted points in (b). (d) The distortion parameters are recovered from these noisy image points using the algorithm based on objective function ξ_3. These parameters are used to undo the distortions present in (b). Despite the large amount of noise, the recovery of undistorted image points is found to be accurate and robust.

Table 1(a): Experimental results for ξ_1.

$\mathcal{L}$	C_3	C_4	P_1	P_2	$w = 0$	$w = 1$	$w = 2$	$w = 5$
#1	10^{-5}	10^{-9}	10^{-5}	10^{-5}	0.000	3.360	13.973	42.521
	10^{-5}	10^{-9}	0.000	0.000	0.000	3.264	13.917	42.574
#2	10^{-5}	10^{-9}	10^{-5}	10^{-5}	0.000	12.095	39.567	66.817
	10^{-5}	10^{-9}	0.000	0.000	0.000	12.184	39.616	66.849

Table 1(b): Experimental results for ξ_2.

$\mathcal{L}$	C_3	C_5	P_1	P_2	$w = 0$	$w = 1$	$w = 2$	$w = 5$
#1	10^{-5}	10^{-9}	10^{-5}	10^{-5}	0.000	0.356	2.473	12.383
	10^{-5}	10^{-9}	0.000	0.000	0.000	0.396	2.272	12.373
#2	10^{-5}	10^{-9}	10^{-5}	10^{-5}	0.000	1.618	5.448	28.639
	10^{-5}	10^{-9}	0.000	0.000	0.000	1.592	5.550	28.711

Table 1(c): Experimental results for ξ_3.

$\mathcal{L}$	C_3	C_5	P_1	P_2	$w = 0$	$w = 1$	$w = 2$	$w = 5$
#1	10^{-5}	10^{-9}	10^{-5}	10^{-5}	0.002	0.363	0.390	0.398
	10^{-5}	10^{-9}	0.000	0.00	0.003	0.328	0.273	0.318
#2	10^{-5}	10^{-9}	10^{-5}	10^{-5}	0.008	0.663	0.773	0.502
	10^{-5}	10^{-9}	0.000	0.000	0.006	0.529	0.734	0.330

Table 1(d): Detailed experimental results for ξ_3.

$\mathcal{L}$	C_3	C_5	P_1	P_2	$w = 0$	$w = 1$	$w = 2$	$w = 5$
#1	10^{-5}	10^{-9}	10^{-5}	10^{-5}	0.002	0.428	0.522	0.391
	10^{-5}	10^{-9}	0.000	0.000	0.004	0.344	0.382	0.246
	10^{-5}	10^{-10}	0.000	0.000	0.281	0.348	0.579	2.818
	10^{-5}	10^{-10}	10^{-6}	10^{-6}	0.007	0.278	0.623	2.782
#2	10^{-5}	10^{-9}	10^{-5}	10^{-5}	0.000	0.151	0.015	0.068
	10^{-5}	10^{-9}	0.000	0.000	0.003	0.305	0.339	0.221
	10^{-5}	10^{-10}	0.000	0.000	0.029	0.152	0.345	1.591
	10^{-5}	10^{-10}	10^{-6}	10^{-6}	0.068	0.192	0.339	1.701
#3	10^{-5}	10^{-9}	10^{-5}	10^{-5}	0.000	0.501	0.574	0.590
	10^{-5}	10^{-9}	0.000	0.000	0.007	0.329	0.330	0.337
	10^{-5}	10^{-10}	0.000	0.000	0.043	0.444	0.488	2.356
	10^{-5}	10^{-10}	10^{-6}	10^{-6}	0.009	0.415	0.645	2.368

Table 2: Results on estimation of optical center (x_p, y_p).

$\mathcal{L}$	C_3	C_5	P_1	P_2	Grid	$w = 0$	$w = 1$	$w = 2$
#1	10^{-5}	10^{-9}	10^{-5}	10^{-5}	2	0.002	4.232	9.014
	10^{-5}	10^{-9}	10^{-5}	10^{-5}	5	0.002	0.363	10.220
	10^{-5}	10^{-9}	10^{-5}	10^{-5}	10	0.002	0.363	0.390
#2	10^{-5}	10^{-9}	10^{-5}	10^{-5}	2	0.008	4.271	3.792
	10^{-5}	10^{-9}	10^{-5}	10^{-5}	5	0.008	0.663	12.017
	10^{-5}	10^{-9}	10^{-5}	10^{-5}	10	0.008	0.663	0.773

image center. The search for the optical center was done using a 5x5 grid and grid resolutions of 10, 5 and 2 pixels. As Table 2 indicates, fine searches in the presence of noise can result in inaccurate solutions, while coarse searches appear to give better results. The time taken to recover all six distortion parameters $(C_3, C_5, P_1, P_2, x_p, y_p)$ is linear in the number of grid points used. Calibration on a 5x5 grid

(a)

(b)

Figure 5: (a) Image produced by a Computar 2.5mm lens and a Computar 1/3″ CCD board camera. (b) Distortion parameters recovered via the minimization of ξ_3 are used to map the output of the camera to perspective video.

took about 20 minutes on a 300MHz Pentium II PC.

We also used ξ_3 to calibrate a 1/2″ Sony XC-75 CCD camera with a Computar 3.6mm lens, and a 1/3″ Computar EMH200-L25 CCD board camera with a 2.5mm lens.

The calibration of the images was done using a set of about 10 lines and a total of about 250 points. The estimated distortion parameters obtained using ξ_3 were used to undistort the images (see Figure 5 for an example). As can be seen from Figure 5(b), straight lines in the scene map to straight lines in the distortion corrected image.

For imaging systems having a large field of view, a perspective projection model may not be appropriate for visualization purposes [Fleck, 1995]. However, the recovery of the distortion parameters facilitates the mapping of the image using any other projection model. For instance, for a wide-angle system, a panoramic projection model or a stereoscopic model may be more suitable.

6 Polycameras

We now describe a natural application for the results we have presented in the paper. A polycamera is a tight cluster of cameras that together capture a large field of view.
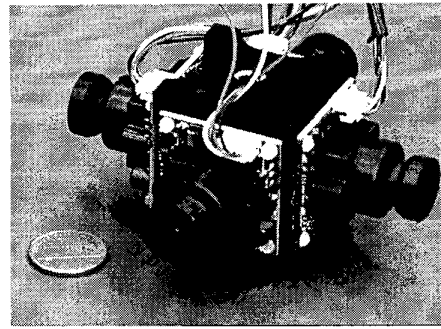

Figure 6: A panoramic polycamera configured using four 1/3″ CCD Computar EMH200-L25 board cameras with 2.5mm lenses. Each camera has a horizontal field of view of about 115 degrees, ensuring a complete field of view of 360 degree.

Unlike multiple cameras used in stereo for instance, the cameras that comprise a polycamera are configured to have minimally overlapping fields of view. In spirit, this idea is similar to that of Nalwa's [Nalwa, 1996] where four cameras and four planar mirrors are configured to obtain a panoramic field of view as seen from a single viewpoint. In our case, we relax the single viewpoint constraint, but ensure that the individual viewpoints of the cameras are close enough that the images they produce can be merged together seamlessly for objects beyond a minimum distance from the polycamera. We call this distance the *minimum working distance*. A more detailed treatment of polycameras and their working distances is provided in [Swaminathan and Nayar, 1999].

Given any desirable field of view, we would like to use the minimum number of cameras to capture it. Clearly, using perspective imaging systems with relatively long focal lengths will necessitate the use of a larger number of cameras. This will require us to acquire and process a large number of video signals as well as increase the working distance of the cluster. In this context, wide-angle imaging systems prove ideal. For instance, the Computar EMH200-L-25 board camera, for which results are presented in Figure 5, has a horizontal field of view of about 115 degrees. Hence, four such cameras can be oriented 90 degrees apart to capture a 360 degree panoramic field of view. In addition, neighboring cameras will have overlapping fields of view, which are necessary to make the complete view seamless. Figure 6 shows the polycamera we have developed based on this design. The complete sensor is enclosed in a cylinder that is 7 cm tall and 7.5 cm in diameter.

We generate the panorama by projecting the individual views of the cameras onto a cylinder. The assumption is that the individual cameras adhere to the perspective projection model. Hence, we first apply our non-metric calibration algorithm to determine the distortion parameters

Figure 7: Panoramic video stream generated using the polycamera shown in Figure 6. The 360 degree panorama is computed using a look-up table, which is constructed taking into account the relative orientations of the four wide-angle cameras as well as their distortion parameters. Notice at extreme right, the extended hand gets warped to the left of the panoramic image.

of each of the four camera to produce four perspective views. Next, corresponding points (produced by distant scene points) in the overlapping fields of view of two adjacent cameras are used to determine the relative orientation (assumed to be purely rotational) between the two cameras. This procedure is applied to all pairs of adjacent cameras.

To account for the differences in the gains of the four cameras, a simple blending algorithm is used in the overlap regions. This algorithm computes the intensity at each pixel within an overlap region as the weighted sum of the intensities in the two view. The weights are proportional to the distances of the pixel from the boundaries of their respective images [Szeliski, 1996]. The mapping from individual views to the panorama as well as the blending weights in the overlap regions are stored as look-up tables for efficiency. This is possible since the relative orientations of the cameras and their distortion parameters do not vary with time.

The four video streams are captured simultaneously using four Matrox boards that reside in a 400 MHz Pentium-II PC. The panorama computed using the above technique is 1000x480 pixels in size and is displayed using Direct-Draw technology at approximately 15 frames a second. Figure 7 shows a snap-shot of the panoramic video produced by this polycamera system.

References

[Becker and Bove, 1995] S. Becker and V.M. Bove. Semi-automatic 3-d model extraction from uncalibrated 2-d camera views. *In Proc. SPIE Visual Data Exploration and Analysis II*, 2410:447–461, Feb 1995.

[Born and Wolf, 1965] M. Born and E. Wolf. *Principles of Optics*. Permagon Press, 1965.

[Brown, 1966] D.C. Brown. Decentering distortion of lenses. *Photogrammetric Engineering*, 32(3):444–462, May 1966.

[Brown, 1971] D.C. Brown. Close range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, Aug 1971.

[Conrady, 1919] A. Conrady. Decentering lens systems. *Monthly notices of the Royal Astronomical Society*, 79:384–390, 1919.

[Fleck, 1995] M. Fleck. Perspective projection: the wrong imaging model. Technical Report 95-01, University of Iowa, Computer Science, 1995.

[Goshtasby, 1989] A. Goshtasby. Correction of image deformation from lens distortion using bezier patches. *Computer Vision, Graphics, and Image Processing*, 47:385–394, 1989.

[Kang, 1997] S.B. Kang. Semi-automatic methods for recovering radial distortion parameters from a single image. *DEC, Cambridge Research Labs, Technical Reports Series CRL 97/3*, May 1997.

[Nalwa, 1996] V. Nalwa. A True Omnidirectional Viewer. Technical report, Bell Laboratories, Holmdel, NJ 07733, U.S.A., February 1996.

[Nelder and Mead, 1965] J.A. Nelder and R.A. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.

[Sawhney and Kumar, 1999] H.S. Sawhney and R. Kumar. True multi-image alignment and its application to mosaicing and lens distortion correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(3):235–243, March 1999.

[Stein, 1995] G.P. Stein. Accurate internal camera calibration using rotation, with analysis of sources of error. In *Proceedings of the 5th International Conference on Computer Vision*, pages 230–236, 1995.

[Stein, 1997] G.P. Stein. Lens distortion calibration using point correspondences. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition*, pages 143–148, San Francisco, June 1997. IEEE Computer Society.

[Swaminathan and Nayar, 1999] Rahul Swaminathan and Shree K. Nayar. Polycameras: Camera clusters for wide angle imaging. Technical Report CUCS-013-99, Columbia University, Computer Science, April 1999.

[Szeliski, 1996] R. Szeliski. Video mosaics for virtual environments. *IEEE Computer Graphics and Applications*, 16(2):22–30, March 1996.

[Tsai, 1987] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision. *International Journal of Robotics and Automation*, 3(4):323–344, Aug 1987.

[Weng et al., 1992] J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(10):965–980, Oct 1992.