Polycameras: Camera Clusters for Wide Angle Imaging *

CUCS-013-99

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

*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.

Abstract

We present the idea of a polycamera which is defined as a tightly packed camera cluster. The cluster is arranged so as to minimize the overlap between adjecent views. The objective of such clusters is to be able to image a very large field of view without loss of resolution. Since these clusters do not have a single viewpoint, analysis is provided on the effects of such non-singularities. We also present certain conbfigurations for polycameras which cover varying fields of view. We would like to minimizze the number of sensors required to capture a given field of view. Therefore we recommend the use of wide-angle sensors as opposed to traditional long focal length sensors. However, such wide-angle sensors tend to have severe distortions which pull points towards the optical center. This paper also proposes a method for recovering the distortion parameters without the use of any calibration objects. Since 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 distortion 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 stream from a wide angle camera can be undistorted in real time using look up tables. Finally, we apply our distortion correction technique to a polycamera made of four wide-angle cameras to create a high resolution 360 degree panorama in real-time.

Index Terms — camera calibration, wide-angle lens, radial distortion, decentering distortion, camera clusters, polycamera, single viewpoints constraint, minimum working distance, real-time panoramic sensor.

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.

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

If the optics of a wide-angle camera system are known *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.

[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.

Several calibration techniques 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, it 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. Becker [1995] used three mutually orthogonal sets of parallel lines and a vanishing point constraint to recover distortion parameters. In [Stein, 1997], [1993] and [1995], point correspondences in multiple images are used to estimate radial distortions. Thus apart from estimating the distortion parameters, one has also to estimate the relative orientation between views. This makes the problem more unstable in the presence of noise. Another novel approach suggested by Sawhney in [1997] and [1999] is, image based distortion parameter estimation. This is a direct method and relies solely on multi-image alignment to estimate the parameters.

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. 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 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. Fewer cameras also facilitate tightly packed clusters, which aid in reducing the effects of a non-singular viewpoint.

In practice, due to the finite size of cameras, it is not possible to maintain a singular viewpoint. One exception being the system developed by Nalwa (see [Nalwa, 1996]), which uses mirrors along with image sensors to obtain a single viewpoint. Non singular viewpoints lead to disparities in the projections of the imaged points, which although useful in stereo applications, do not help in creating smooth mosaics. Although theoretically this disparity vanishes only for points at infinity, we propose the idea of a *minimum working distance* beyond which the disparity falls below a detectable threshold. Analysis of the minimum working distance for the generic case, as well as results for a panoramic polycamera are included. Finally, results of a real time high resolution panoramic sensor we built are presented.

2 Distortion Model

Distortions in lenses can be decomposed into three components: (a) shift of the optical center, (b) radial distortion, and (c) decentering distortion.

Let the perspective projection of a scene point be $\mathbf{q}'$ (see Figure 2). Due to distortions in the lens, $\mathbf{q}'$ gets mapped to $\mathbf{q}$. Let (x, y) be the Cartesian and (r, ϕ) be the polar coordinates of $\mathbf{q}$. Similarly, let (x', y') be the Cartesian and (r', ϕ') be the polar coordinates of $\mathbf{q}'$. 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:

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

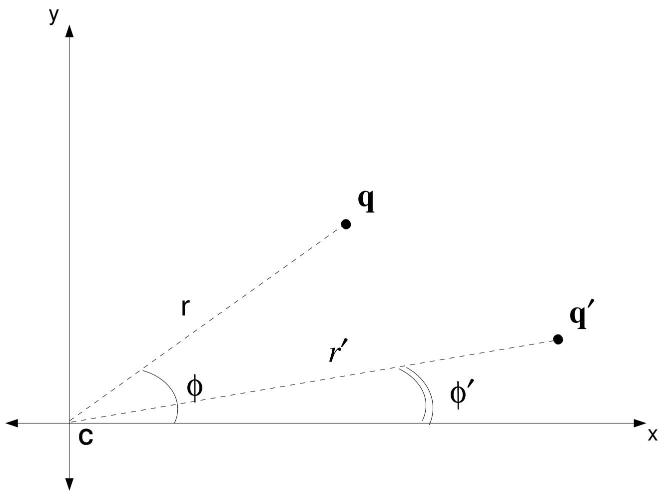


Figure 2: $\mathbf{q}'$ is the perspective projection of a scene point onto the image plane. Due to radial and decentering distortions, $\mathbf{q}'$ gets mapped to the point $\mathbf{q}$.

2.1 Shift of Optical Center

A shift of the optical center corresponds to a shift of the image detector in a plane perpendicular to the optical axis. The effect of such a distortion is merely that the image center is no longer the optical center. Estimating this distortion component amounts to estimating $C = (x_p, y_p)$, the optical center.

2.2 Radial Distortions

There are two kinds of radial distortions in lenses. The one found in most wide angle cameras tend to pull points radially towards the optical center. This is also referred to as *barrel distortion* [Born and Wolf, 1965]. The other type of distortion tends to push points away from the optical center along the radial direction and is called *pin-cushion distortion*. As can be seen, these effects are purely radial in direction and solely depend on the distance from the optical center. The radial distortion present in the point $\mathbf{q}$ can be written as:

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

where, C_{2i+1} are the distortion parameters. As can be seen, only the odd powered terms are used to model this distortion [Conrady, 1919]. The higher powered terms tend to contribute lesser to the effective distortion. We therefore ignore terms higher than the fifth-order as their contribution to the distortion is negligible in practice [Brown, 1966]. Hence, we have:

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

2.3 Decentering Distortions

Decentering distortions are caused by the non-orthogonality of the lens components and the image detector with respect to the optical axis. All imaging systems have some decentering distortions. Unlike radial distortions, this distortion component acts tangential to the radial direction. We use Conrady's model [Conrady, 1919] for decentering distortion:

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

where, P_i are the distortion parameters and $\Delta T_x, \Delta T_y$ are the distortions along the x and y directions, respectively.

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

$$\begin{aligned} \Delta T_x(\mathbf{q}) &\approx [P_1 r^2(1 + 2\cos^2(\phi)) + 2P_2 r^2 \sin(\phi)\cos(\phi)] \\ \Delta T_y(\mathbf{q}) &\approx [P_2 r^2(1 + 2\sin^2(\phi)) + 2P_1 r^2 \sin(\phi)\cos(\phi)] \,. \end{aligned} \tag{5}$$

2.4 Complete Distortion Model

The total distortion is modeled as a combination of the above three components:

$$\begin{aligned} \Delta x(\mathbf{q}) &\approx \cos(\phi)[\Delta r(\mathbf{q})] + \Delta T_x(\mathbf{q}) \\ \Delta y(\mathbf{q}) &\approx \sin(\phi)[\Delta r(\mathbf{q})] + \Delta T_y(\mathbf{q}) \,. \end{aligned} \tag{6}$$

In order to correct distortions, we thus 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. Consider a set of points in the scene which lie on a straight line. Their projection onto the image lie on 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 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. These selected points will be the used to estimate the distortion parameters.

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 $\{\mathbf{q} = (x, y)\}$. This maps the points to newer points which we hope will be collinear. Lines are fitted to the resulting sets of supposed collinear points $\{\mathbf{q}' = (x', y')\}$ using a least squares approach. The objective function is then defined as the sum of the squared distances of the points from their corresponding "best-fit" lines (see Fig. 3).

Let the points selected by the user along any curve be denoted by the set $\{\mathbf{q}\}$. On applying the

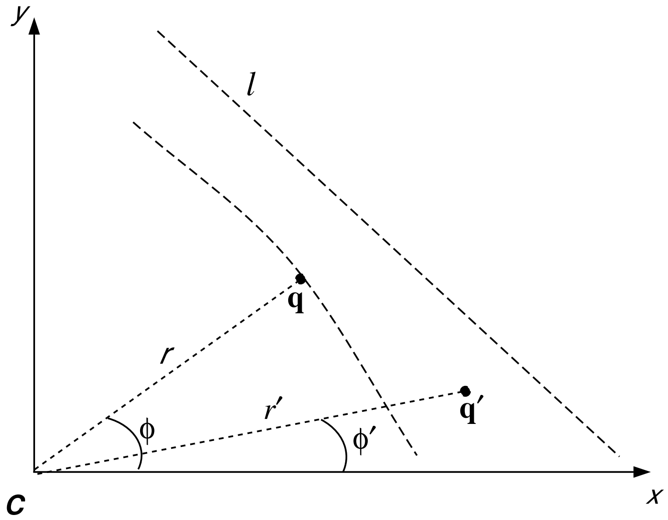


Figure 3: $\mathbf{q}$ is a point selected along the image curve. Applying the current set of hypothesized distortion parameters $\mathcal{S}$, we get its undistorted location at $\mathbf{q}'$. $\mathbf{l}$ represents the best fit line for the set of undistorted points $\{\mathbf{q}'\}$. We wish to minimize the distance between $\mathbf{q}'$ and $\mathbf{l}$.

hypothesized distortion parameters these points get mapped to the set $\{\mathbf{q}'\}$. Let the best-fit line for a set of points $\{\mathbf{q}'\}$ 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 point $\mathbf{q}$ is defined as:

$$e = (x' \sin(\theta) - y' \cos(\theta) + \rho)^2 \ ,$$

where:

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

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:

$$\xi_1 \ = \ \sum_{l=1}^{L} \sum_{p=1}^{P_l} \left(x'_{p,l} \sin(\theta_l) - y'_{p,l} \cos(\theta_l) + \rho_l \right)^2 \ , \tag{8}$$

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 p^{th} 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. The modified objective function then is:

$$\xi_2 \quad = \quad \sum_{l=1}^{L} \sum_{p=1}^{P_l} \left(\frac{x'_{p,l} \sin(\theta_l) - y'_{p,l} \cos(\theta_l) + \rho_l}{\rho_l} \right)^2 . \tag{9}$$

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 $\mathcal{S}$). Also the distortion model nonlinearly magnifies noise in the undistorted domain. Thus an error of a few pixels in the distorted point space could easily map to an error of hundreds of pixels in the undistorted space. It is therefore more appropriate to formulate an objective function that uses errors computed in the space of distorted image points.

As shown in Figure 4, 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, as before, l is the best-fit line for 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 $\hat{\mathbf{q}}'$ (see Figure 4). The new error function is defined as:

$$e = \|\mathbf{q} - \hat{\mathbf{q}}\|^2 . \tag{10}$$

Since $\hat{\mathbf{q}}'(\hat{x}', \hat{y}')$ must lie on l, it must satisfy the constraint:

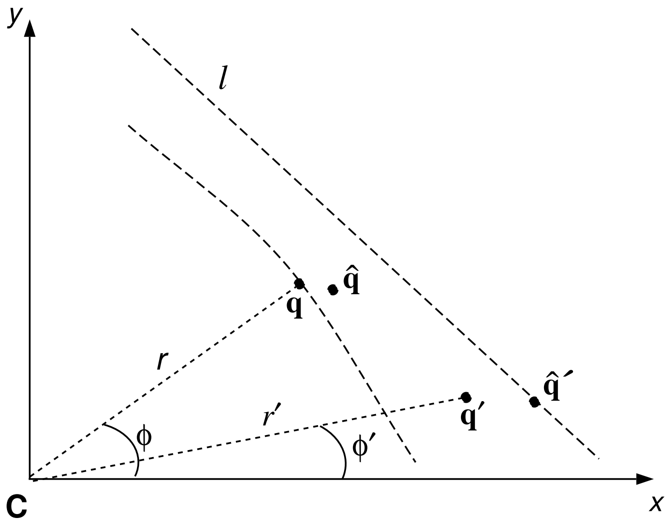


Figure 4: $\mathbf{q}$ is a point selected by the user and $\mathbf{q}'$ is its undistorted location on applying the (hypothesized) distortion parameters $\mathcal{S}$. l is the "best-fit" line estimated for all $\mathbf{q}'$, which lie on the same scene line. $\hat{\mathbf{q}}$ is a point close to $\mathbf{q}$ such that, its undistorted location $\hat{\mathbf{q}}'$ (obtained by applying $\mathcal{S}$ on $\hat{\mathbf{q}}$) lies on l. We wish to minimize the distance between $\mathbf{q}$ and $\hat{\mathbf{q}}$.

$$\hat{x}' \sin(\theta) - \hat{y}' \cos(\theta) + \rho = 0 \ ,$$

where:

$$\hat{x}' = \hat{x} + \Delta x(\hat{\mathbf{q}}) \ , \ \hat{y}' = \hat{y} + \Delta y(\hat{\mathbf{q}}) \ . \tag{11}$$

Using all the selected points, the objective function is determined as:

$$\xi_3 \ = \ \sum_{l=1}^{L} \sum_{p=1}^{P_l} \|\mathbf{q}_{p,l} - \hat{\mathbf{q}}_{p,l}\|^2 \ . \tag{12}$$

We have found this objective function to be much more robust in the presence of noise. Experimental results included in the following sections illustrate this fact.

4 Minimization of ξ_1, ξ_2 and ξ_3

We now describe the non-linear search algorithms used to recover the distortion parameters $\mathcal{S}$ 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. The nonlinear search method also needs a starting seed point to begin its search. We assume that there are no distortions present initially and hence all the distortion parameters are set to zero.

At each step of the non-linear search, given the set of (hypothesized) parameters $\mathcal{S}$, 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 point $\hat{\mathbf{q}}$ (see (12)), for which there is no closed-form solution.

Hence, we solve for each $\hat{\mathbf{q}}_{p,l}$ by searching the neighborhood of $\mathbf{q}_{p,l}$ for the point which, when undistorted using $\mathcal{S}$, lies on l. This requires a $2D$ search (see Figure 4), which is computationally intensive. For efficiency, we use a $1D$ search along the radial direction, since there always exists a point in the radial direction of the selected point $\mathbf{q}$, which lies on the true distorted curve. This approximation enables a faster estimation of the distortion parameters $\{C_3, C_5, P_1, P_2\}$. Our current implementation takes under 30 seconds on a 300MHz Pentium II PC to estimate the radial and decentering distortion parameters.

Note, however, that we did not include the optical center $C(x_p, y_p)$, in the non-linear search for the distortion parameters $\mathcal{S}$. 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. This was also observed by Brown [1971], even though the severity of distortions as well as the noise levels were much lower. Therefore, we recommend nesting 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).

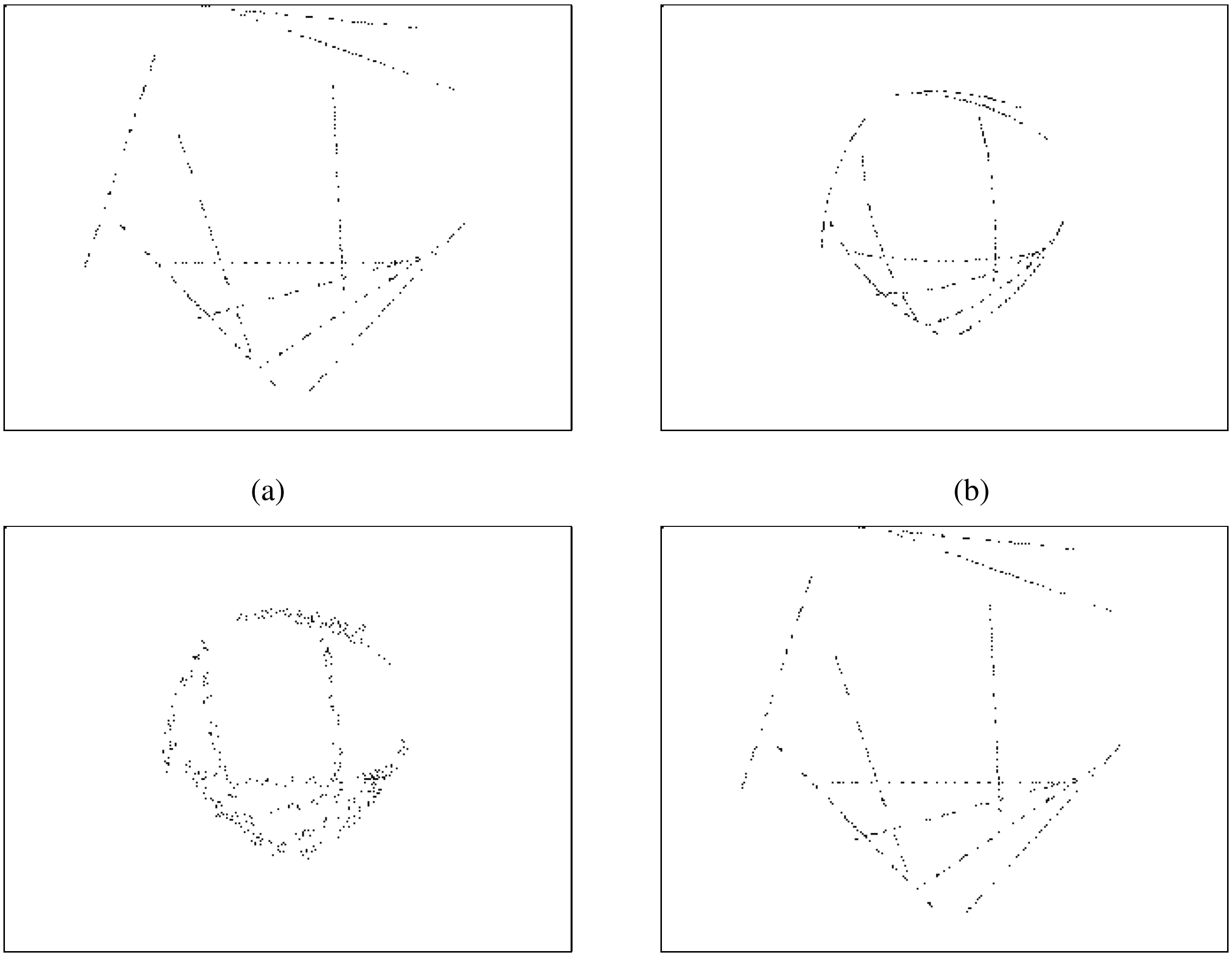
5 Synthetic Experiments

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 simulate the imaging process including the distortions to quantitatively evaluate the robustness of the method.

Points were randomly sampled from synthetically generated lines with random orientations and positions (see Figure 5(a)). Using known distortion parameters, the sampled points were distorted (see Figure 5(b)). To simulate erroneous point selection, we added uniform noise in the interval $(-w, +w)$ to the points (see figure 5(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 5(d)).

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

Tables 1 (a)-(c) 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.



(a) (b)

(c) (d)

Figure 5: (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.

In contrast ξ_3 is much more robust and can yield 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 2. The interesting fact to be noted is that even for large levels of input noise $(w = 5)$, the resulting average error is below 5 pixels. This is interesting as the undistorting process tends to exaggerate all errors.

Distortion Coefficients					Average Error (pixels)			
$\mathcal{L}$	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

(a)

Distortion Coefficients					Average Error (pixels)			
$\mathcal{L}$	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

(b)

Distortion Coefficients					Average Error (pixels)			
$\mathcal{L}$	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

(c)

Table 1: Results on running simulations on the three objective functions: (a) Indicates errors on undistoring the image samples on using ξ_1. (b) Results for ξ_2. More robust to noise but yet degrades for large noise levels. (C) Results on using ξ_3. Note the striking improvement over both ξ_1 and ξ_2

Distortion Coefficients					Average Error (pixels)			
$\mathcal{L}$	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: Detailed experimental results for ξ_3.

Distortion Coefficients						Average Error (pixels)		
$\mathcal{L}$	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

Table 3: Results on estimation of optical center (x_p, y_p).

As mentioned earlier, recovery of the optical center is implemented as a coarse-to-fine exhaustive search around the image center. The search for the optical center was done using a 5x5 grid at resolutions of 10, 5 and 2 pixels. As Table 3 indicates, fine searches in the presence of noise can result in inaccurate solutions, while coarse searches appear to give better results. It is not surprising that the search for the optical center is unstable in the presence of decentering distortions as was previously observed by Brown [1971]. 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. The run time for the complete calibration algorithm for a 5x5 grid is about 20 minutes on a 300 MHz Pentium II machine.

6 Results with Real Images

We tested our algorithm based on ξ_3 using images taken with two different camera systems. To test robustness over a wide range of conditions we used a low distortion camera as well as an inexpensive board camera with severe decentering and radial distortions. The low distortion sensor is a $1/2''$ CCD Sony XC-75 camera with a Computar 3.6mm lens and the high distortion sensor is a $1/3''$ Computar EMH200-L25 CCD board camera with a 2.5mm lens.

The calibration of the sensors 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 distorted images (see figures 6 (a, c) for an example). As can be seen from figures 6(b, d) , straight lines in the scene map to straight lines in the distortion corrected image.

Furthermore, we can construct lookup tables for each sensor that can be used to generate an undistorted image stream in real-time. Since the lookup table essentially represents an image warping function, we have to address issues relating to speed and image quality. The warping function tends to scale up the image in a nonlinear fashion. This increase in scale factor causes certain pixels in the output image frame to map to sub pixel locations in the source image. We use nearest neighbor interpolation here as this is the fastest in terms of real-time performance.

The lookup table uses backward mapping and hence we need to find for each output pixel, its

(a)

(b)

(c)

(d)

Figure 6: (a) Image captured with a Computar 3.6mm lens and a Sony $1/2''$ CCD camera (b) Distortion parameters recovered via the minimization of ξ_3 are used to map (a) to a perspective image. (c) Image produced by a Computar 2.5mm lens and a Computar $1/3''$ CCD board camera. (d) Distortion parameters recovered via the minimization of ξ_3 are used to map (c) to perspective image. Notice that, straight lines in the scene, such as door edges, map to straight lines in the undistorted images.

source from the distorted image frame. Since there does not exist a closed form solution for the inverse of the distortion model, we need to search in a nonlinear fashion for the image points which would act as the source for the corresponding undistorted point. This process although time consuming needs to be performed only once. Then on, the lookup table can be used to generate an undistorted video stream in real time.

7 Polycameras

We now describe a natural application for the results we have presented so far in the paper. We define a polycamera to be a tight cluster of cameras that together capture a large field of view. Unlike multiple cameras used in stereo for instance, the cameras that comprise a polycamera are configured to have minimally overlapping fields of view. Due to the finite size of the cameras it is difficult to configure them so as to have a single viewpoint. We therefore relax the single viewpoint constraint, but ensure that the individual viewpoints of the cameras are close enough so 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* of the polycamera.

In spirit, the idea of using multiple sensors 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. The singular viewpoint is achieved by configuring the mirrors and cameras such that their centers of projection reflect to the same $3D$ point. This point thereof being the center of projection for the panoramic sensor. However, this requires careful arrangement of the mirrors and the cameras, which is avoided in our system.

Another sensor that also uses camera clusters to capture a wider field of view, is that developed by Immersive Media [Media, 1999] called the *Dodeca*. The Dodeca unlike the sensor developed by Nalwa, does not use any mirrors. Instead, it uses 11 image sensors arranged on a sphere, tessellated as a dodecahedron. Each sensor has a small field of view and a long focal length. However, small field of view sensors, necessitates the use of a large number of such sensors. The effect of this is a strong deviation from the single viewpoint constraint, thereby increasing the minimum working distances of the cluster. Also, larger number of sensors necessitate the capability of acquiring and processing more video signals.

Given a desirable field of view, we would like to use the least 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, as is the case with the Dodeca. We therefore propose the use of wide-angle imaging systems. Typically, such wide angle sensors have radial and tangential distortions. For geometrically correct projections and smooth blending between views, we first have to calibrate the sensors for the distortion parameters. We recommend the calibration method proposed earlier in the paper, to estimate these parameters. Once calibrated, these sensors can be used to capture large fields of view.

7.1 Polycamera configurations

The wide fields of view of image sensors like for instance the Computar EMH200-L-25 board camera are most suitable for polycameras. This sensor has a horizontal field of view of about 115 degrees. We present here configurations in which such wide-angle sensors can be arranged to provide panoramic, hemispherical and spherical fields of view.

We can arrange six sensors on the sides of a cube, so as to capture a spherical field of view about the center of the cube. The orientation of each sensor on the cube's surface is such that the horizontal and vertical axis of each sensor are orthogonal to those of the adjacent cameras. We assume that that the sum of the horizontal and vertical fields of view is greater than 180 degrees. This ensures an overlapped view between any two edges of the cube (see figure 7 (a)). The final projection could be on a cube or on a sphere. The advantage of this configuration over the Dodeca (see [Media, 1999]), is the use of fewer sensors while still covering the entire sphere. Smaller fewer sensors facilitate building a tighter cluster, thus reducing the parallax effect due to non-singular viewpoints.

Four such sensors can be oriented 90 degrees apart to capture a 360 degree panoramic field of view. The 115 degree horizontal FOV facilitates adjacent views to have overlapped regions of space of upto 25 degrees. In general any sensor which has a horizontal field of view of over 90 degrees can be used in such a configuration (see figure 7 (b)), to capture a panoramic field of view.

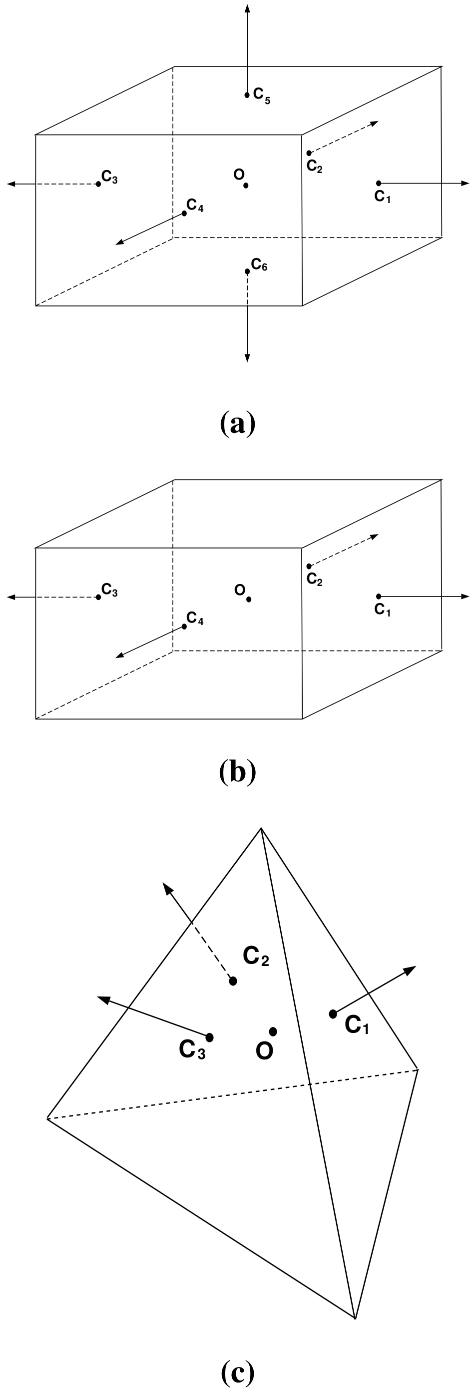


Figure 7: (a) A Spherical FOV polycamera configured using six wide angle sensors. Each sensor is located on the sides of a cube, such that their axis are mutually orthogonal. (b) A panoramic polycamera configured using four wide angle sensors. Each camera has a horizontal field of view of over 90 degrees, ensuring a complete field of view of 360 degree.(c) Three wide angle sensors arranged on the sides of a pyramid. In the above three configurations, the normals on the surface indicate the viewing direction of each sensor.

This configuration uses 3 image sensors arranged on the sides of a pyramid. Depending on the filed of view of the individual sensors, a complete hemisphere to a partial hemisphere may be captured. The advantage of this configuration is that if a single channel sensor is used, then the three signals can be coupled together as one 3-channel signal. This facilitates the use of a single frame grabber thus reducing the computational power needed to acquire and process the signals. Figure 7 (c) illustrates this configuration.

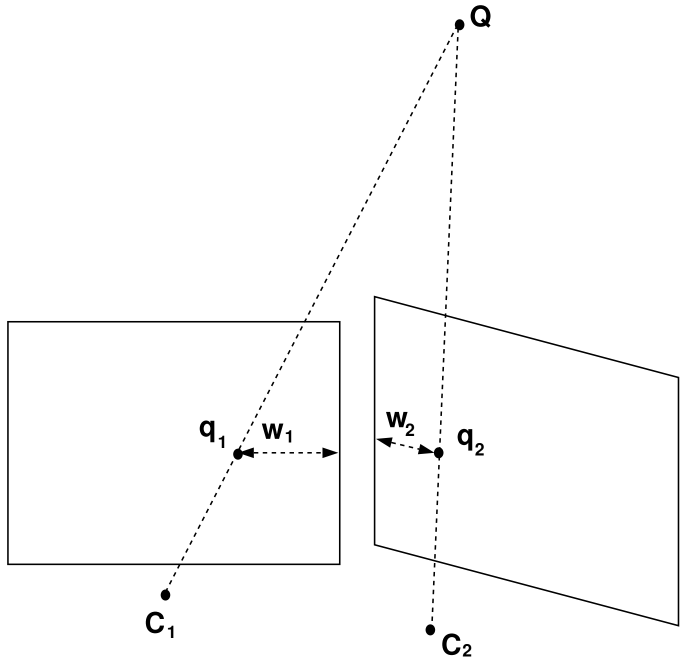


Figure 8: Schematic showing two cameras centered at C_1 and C_2, imaging the scene point Q. the point Q is imaged at the points q_1 and q_2, respectively. The distances to the closes edge of the corresponding images are then w_1 and w_2 respectively. These distances are used to weight the contribution of a pixel's intensity.

7.2 Blending multiple views

When stitching together multiple views, it is possible for more than one view to contribute to a single pixel along view boundaries and regions of overlap. This can be handled by either arbitrarily selecting any one view to be the source or by more sophisticated methods. The first, more simpler method, could result in severe visual artifacts due to the different gains of the sensors, across such view boundaries. We therefore utilize a method suggested by Szeliski [1996] for blending the multiple views with seamless transitions between views. The intensity of the resulting pixel is defined as the weighted sum of the intensities of the pixels in the respective views. The weights are based on the closeness of the pixel to the view's image boundary. The closer a pixel is to its boundary the lesser is its contribution. This gives seamless transitions between views.

Consider the case of N sensors imaging a scene point Q. Let any sensor C_i image the point Q at q_i. Let w_i be the corresponding shortest distance to any edge of the image (see figure 8). If the intensity of the imaged point q_i is given by I_{q_i}, then the resulting image intensity on the projection surface is:

$$I = \frac{\sum_{i=0}^{N}(w_i \cdot I_{q_i})}{\sum_{i=0}^{N}(w_i)} \tag{13}$$

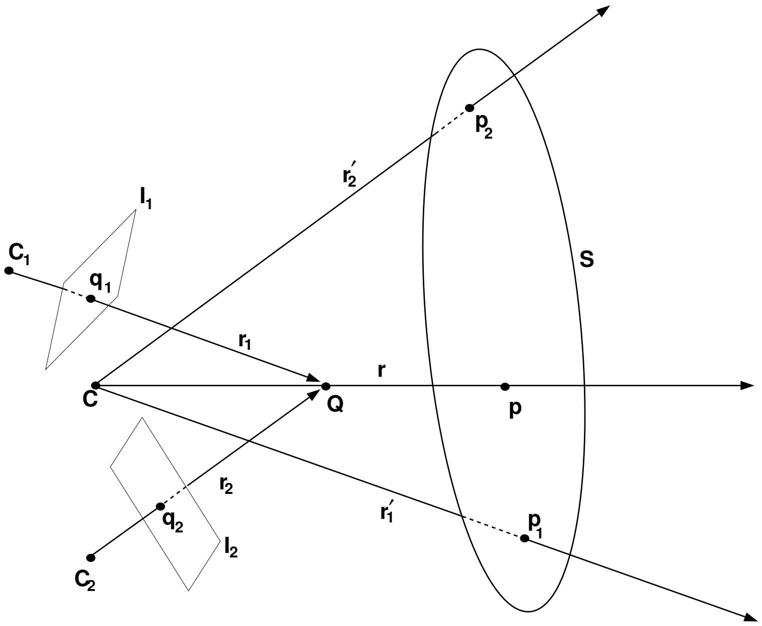


Figure 9: Two cameras C_1 and C_2, image the scene point Q at points q_1 and q_2, respectively. Their projections onto the surface S (assuming pure rotation between views) are p_1 and p_2 respectively. p_1 and p_2 are obtained by intersecting rays parallel to r_1 and r_2, denoted as r_1' and r_2' with S. Q should ideally project onto S at p. The distance of Q from O, at which the disparity between p_1 and p_2 fall below some threshold is the *minimum working distance*.

Figure 10: Segment of panorama clearly showing disparity in image points for close up objects. The ghosting effect within the dotted region, is due to the scene points being close and at a distance lesser than the minimum working distance. The disparity is very conspicuous at the corner of the monitor .

8 Non-Singular viewpoints and the Minimum Working Distance

Projection onto any surface requires a single center of projection O. The finite size of each individual sensor causes the polycamera to have a non-singular viewpoint. Due to the sampling nature of the sensors, there is a minimum distance which we call "minimum working distance", beyond which the non singularity of viewpoints has negligible effect. Below this distance, disparities between projections from multiple views would be large. Figure 9 shows the disparity of a point Q when projected from two views, centered at C_1 and C_2. Blending multiple views of such scene points, results in a ghosting effect which is clearly demonstrated in figure 10. It should however be noted that the working distance depends on the resolution of the projection surface. Thus reducing resolution will result in a decrease of the minimum working distance.

We now consider the problem of estimating the minimum working distance given a cluster of cameras. The issue is pertinent when we try to blend the various views onto a single surface of projection. At the minimum working distance we would like the disparity between projections from pairs of views to lie below some preset threshold level (ϵ). Thus for more than two sensors imaging the same point in space, we define the working distance as the maximum distance on considering all sensors.

Consider a point on the projection surface $p = (x, y)$. This corresponds to a ray direction given by the pan and tilt angles (ϕ, θ). For a point Q in space along this ray to have negligible disparity, it should lie at a distance d which is the minimum working distance from O. In rectangular co-ordinates the point Q is given by:

$$\mathbf{Q} = \begin{pmatrix} X & Y & Z & 1 \end{pmatrix}^T = \begin{pmatrix} d \cdot \cos(\theta) \cdot \cos(\phi) \\ d \cdot \sin(\theta) \\ d \cdot \cos(\theta) \cdot \sin(\phi) \\ 1 \end{pmatrix} \tag{14}$$

Consider a camera C_i in whose field of view Q lies. If we represent it's camera matrix (extrinsic parameters such as rotation and translation) by D_i, and let P_i be the perspective projection matrix. Then camera C_i images Q at:

$$q_i = P_i \cdot D_i \cdot Q \tag{15}$$

Let $\mathcal{P}(\cdot)$ be a mapping of a point on the image onto the surface of projection (see figure 9). Thus the imaged point q_i would be projected onto the surface at:

$$p_i = \mathcal{P}(q_i) \tag{16}$$

Disparity between the projections of two views is guaranteed to lie below ϵ by the constraint:

$$\|p_i - p\| = \frac{\epsilon}{2} \tag{17}$$

The distance d which satisfies the above constraint is the minimum working distance along the ray direction (ϕ, θ). To estimate the minimum working distance for the entire cluster, we need to estimate d for all ray direction which project onto the surface. The maximum distance along all ray directions represents the bounding sphere of minimum working distance.

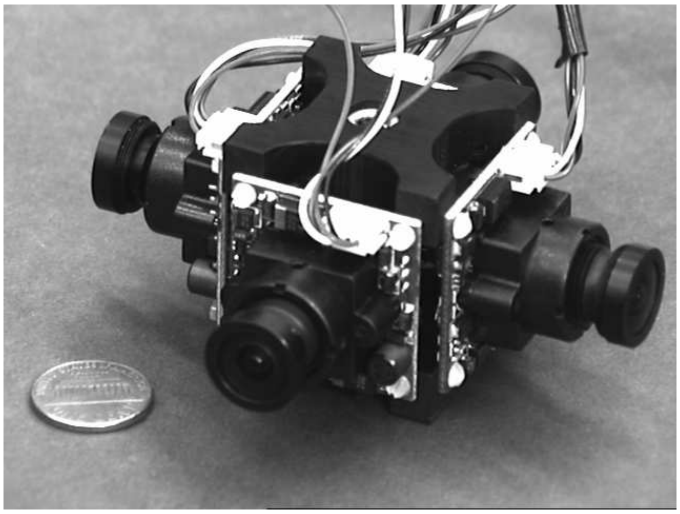


Figure 11: A panoramic polycamera configured using four $1/3''$ CCD Computar EMH200-L25 board cameras with 2.5mm lenses. Each camera has a horizontal field of view of about 115 degrees, ensuring a complete field of view of 360 degree.

9 A Real-time Panoramic Polycamera

Figure 11 shows the polycamera we developed based on design described earlier. The complete sensor is enclosed in a cylinder that is 7 cm tall and 7.5 cm in diameter. It houses 4 Computar EMH200-L25 board cameras with 2.5mm lenses placed approximately 90 degrees apart. Each camera has about 115 degree field of view horizontally and thus ensures overlap between adjacent views as well as a complete 360 degree field of view.

9.1 Minimum working Distance

Figure 12 illustrates the minimum working distance varying across a panorama generated using the four camera panoramic polycamera. Larger working distances are represented by brighter points and nearer distances by darker points. As expected, the distance is minimal along the camera's optical axis and increases as we progress along any direction tangential to this axis. There is a sharp change of the working distance at the overlapped regions on the panorama as we consider the maximum of the two working distances in regions of overlap.

This simulation was run assuming the four sensors were in the same plane and precisely 90 degrees apart. Each was assumed to be displaced from the center of projection of the panorama by $0.01m$.

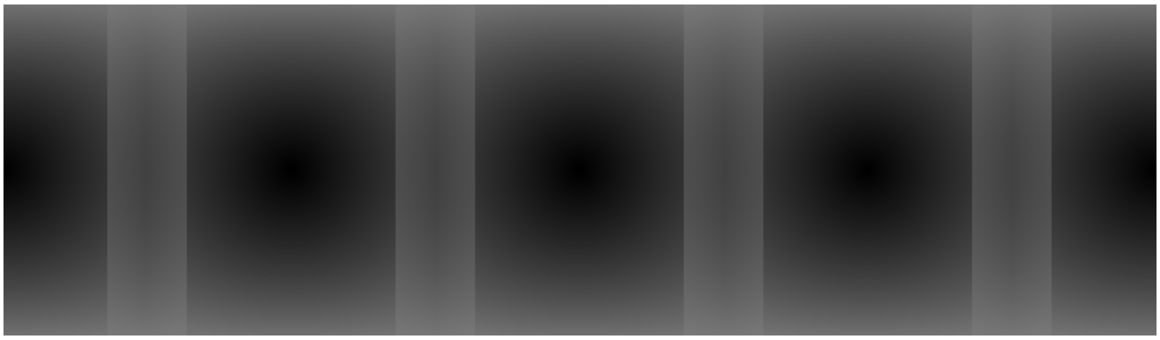


Figure 12: The minimum distance map. This map shows the minimum distance a way from the center of the panorama (cylinder) that a point in space has to be f or a projection error of less than half a pixel. Brighter regions correspond to regions on the panorama where the point has to be further away.

The minimum working distance was estimated to be approximately $4.0m$.

9.2 Panorama generation

Prior to blending adjacent views we need to undistort the image streams. We therefore calibrate each camera using the proposed method (objective function ξ_3). The estimated distortion parameters are used to create look-up tables for the 4 sensors which map points from the undistorted

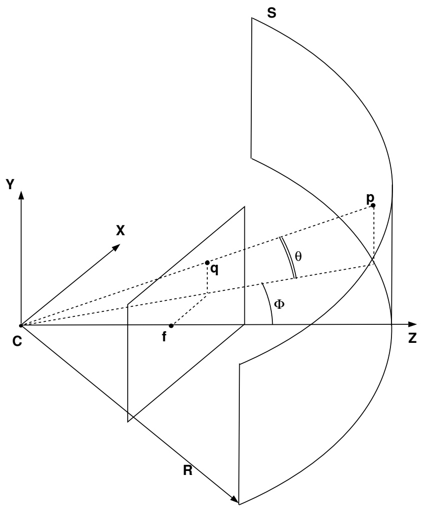


Figure 13: The Panoramic projection model. A point $\mathbf{q}$ in the image is projected along the ray, originating at the center of projection $\mathbf{C}$. The intersection of this ray direction with the cylinder $\mathbf{S}$ at $\mathbf{p}$, represents its panoramic projection. ϕ and θ are the pan and azimuthal angles estimated from $\mathbf{q}$. $\mathbf{R}$ is the radius of the cylinder.

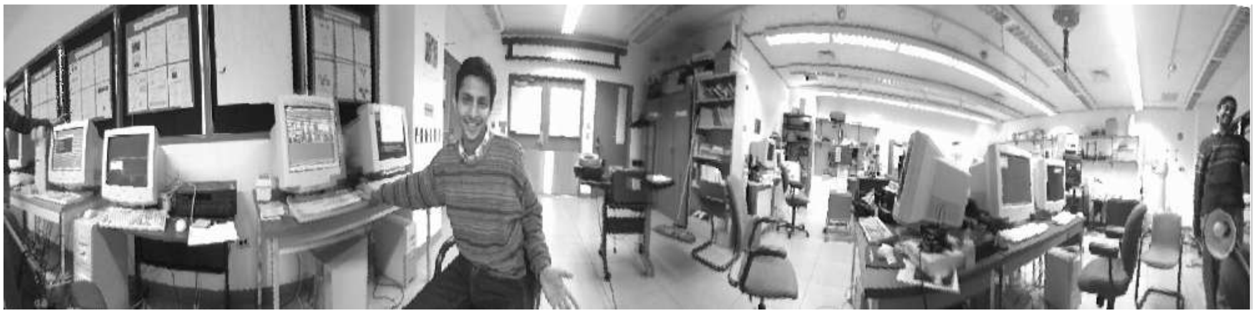


Figure 14: Panoramic video stream generated using the polycamera shown in Figure 11. The panorama is computed using a look-up table, which is constructed taking into account the relative orientations of the four wide-angle cameras as well as their distortion parameters.

space to the source (distorted) space. Point correspondences in overlapping views are used to estimate relative orientation (assumed to be purely rotational) between two sensors. Each pixel on the panorama, represented by a cylinder (see figure 13), maps to at most two camera views. This mapping being time invariant, needs to be computed only once. We again use a look-up table to represent the panoramic projection.

To account for the differences in the gains of the four cameras, the blending algorithm described earlier was used. The mapping from individual views to the panorama as well as the blending weights in the overlap regions are stored in the look-up table. Four video streams are captured simultaneously using four Matrox boards that reside in a 400 MHz Pentium-II PC. The look-up table us used to create a 1000x480 panorama. Displayed of the panorama is done using Direct-Draw technology at approximately 15 frames a second. Figure 14 shows a snap-shot of the panoramic video produced by this polycamera system.

10 Conclusion

In this paper we have proposed a new method to calibrate imaging systems for radial and decentering distortions. Most of the prior methods either relied on calibration objects or on virtually noiseless data. We propose a method that is much more robust in the presence of noise. The constraint used in our approach is that straight lines in the scene should map to straight lines in the image if perspective projection is assumed. The only requirement of the algorithm, is that the user indicate which points in the image, lie on straight lines in the scene. This requirement is easy to satisfy as straight lines are abundant in urban scenes.

The method uses an objective function, which when minimized, yields the lens distortion parameters. This objective function has been defined in the space of distorted points as opposed to defining it in the undistorted space, so as to minimize any nonlinear exaggeration of errors due to noise. This makes it robust to high levels of noise.

Synthetic as well as experiments with real images are provided to demonstrate the robustness of this technique. Noise levels of upto 5 pixels have been simulated and results indicate low error levels in parameter estimation under these circumstances (see tables 1(c), 2 and 3).

We also have proposed the idea of a *polycamera* which we define as a closely packed camera cluster. Due to the finite size of each sensor it is very difficult to adhere to the single viewpoint constraint. We can however relax this constraint due to the sampling nature of the image detector. Beyond some distance from the camera cluster, the effect of a non-singular viewpoint falls below a detectable threshold. We define this distance to be the "minimum working distance". Analysis of the minimum working distance and its estimation for the generic case of a multi-sensor cluster is provided. We also propose certain configurations in which polycameras could be constructed so as to capture various large fields of view ranging from panoramic to a spherical field of view.

Finally, results are presented for one such panoramic polycamera built using four wide angle sensors. Simulating the camera cluster, the estimated minimum working distance has been found to be about 4.0 meters. Since each sensor used in this polycamera has severe distortions, we first estimate the distortion parameters using the proposed method. The undistorted image streams are then projected onto a panorama in real time to generate a 360 degree panorama at 15Hz. This high speed processing is possible by using precomputed look-up tables, and DirectDraw technology for fast rendering of the panorama.

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.

[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.

[Media, 1999] Immersive Media. *Web site: http://www.immersivemedia.com*. 1999.

[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, 1997] H.S. Sawhney and R. Kumar. True multi-image alignment and its application to mosaicing and lens distortion correction. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition*, pages 450–456, 1997.

[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, 1993] G.P. Stein. Internal camera calibration using rotation and geometric shapes. In *MIT AI-TR*, 1993.

[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.

[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.