A Variational Approach to Problems in Calibration of Multiple Cameras

Gozde Unal
Siemens Corporate Research
Princeton, NJ 08540 gozde@scr.siemens.com

Anthony Yezzi
Georgia Institute of Technology
Atlanta, GA 30332 ayezzi@ece.gatech.edu

Abstract

This paper addresses the problem of calibrating camera parameters using variational methods. One problem addressed in this paper is the severe lens distortion in wide angle/inexpensive camera lenses. The camera distortion effects lead to inaccurate 3D reconstructions and geometrical measurements if not accounted for. A second problem is the color calibration problem caused by variations in camera responses which results in different color measurements and affects the algorithms that depend on these measurements. We present multi-view stereo techniques based on variational ideas to address these calibration problems. To reduce computational complexity of such algorithms, we utilize a prior knowledge on the calibration object which is used in the process, and evolve the pose, orientation, and scale parameters of such a 3D model object. We derive the evolution equations for the distortion coefficients, the color calibration parameters of the cameras, and present experimental results which demonstrate their potential use.

1. Introduction

Camera calibration refers to the problem of finding the mapping between the 3D world and the camera or image plane. For most computer vision algorithms aimed at reconstructing reliable digital 3D representations of scenes from multiple 2D images, accurate camera calibrations are essential. There has been a great deal of research on camera calibration problem as early as in 70's [1]. In most of the previous techniques, some set of features are extracted from images of a known calibration pattern, and intrinsic camera parameters as well as camera pose and orientation (extrinsic camera parameters) are estimated by a minimization of an overall cost functional [2–8]. Many calibration techniques use both nonlinear minimization and closed form solutions as in [9]. In this paper, we only address estimation of one of the intrinsic camera parameters, namely lens distortion, and also estimation of camera color calibration parameters in a coupled way within a multiple camera system. The ideas we present, however, may be extended to joint estimation of extrinsic camera parameters as well. The other main intrinsic camera parameters i.e. focal length and the principal point, may be calibrated individually for each camera by one of the existing techniques cited above, or by similar variational ideas presented.

1.1. Relation to Previous Work/Contributions

1.1.1 Lens Distortion

The linear model in an ideal pinhole camera model leads to imaging of world lines as lines on the image plane, and simplifies many computations and considerations [3]. However, for most real cameras with wide-angle or inexpensive lenses this assumption does not hold, and nonlinearities introduced by a well-known phenemonen referred to as a lens distortion should be taken into account. The corresponding distortion parameters should be estimated for each camera.

In many existing calibration techniques, good estimates for external and internal camera parameters are first obtained by a pinhole camera model neglecting lens distortion. Then distortion calibration is performed while holding the other parameters fixed [10–12]. This is possible because the mapping from 3D world coordinates to the 2D image plane can be decomposed into a perspective projection and a mapping that models the deviations from the ideal pinhole camera.

A popular group of lens distortion calibration methods in the literature, mainly under the category known as plumb line methods, rely on a first step of extracting edges from the images. Either a user manually selects the image curves or there must be a way to reliably estimate image edges which correspond to linear 3D segments in the world. An optimization problem is set up by defining a measure of how much each detected segment is distorted. The curved lines in the image which do not really correspond to 3D line segments will constitute outliers in this optimization procedure [10, 13–16]. Other techniques such as [17] rely on point correspondences. Given a set of 3D points, the associated epipolar and trilinear (among three cameras) constraints are arranged into a tensor, which is computed with estimated distortion parameters at each step to minimize a reprojection error in an iterative manner. In another group of methods as in [18–20] a direct solution strategy is employed to find camera calibration parameters by incorporating lens distortion as well.

Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04)
1063-6919/04 $20.00 © 2004 **IEEE**

Our contribution is a new distortion calibration technique which does not rely on extraction of edges and search for point correspondences. The former may not be an easy task due to noise and local extrema. Instead, we devise an integrated calibration technique in which the distortion parameters of cameras are computed in a tightly coupled framework. The desired coupling of multiple camera views comes from estimating a common 3D object (in this case the calibration object). In other words, we minimize the cost between the reprojection of the 3D calibration object and the image measurements by evolving the distortion parameters of the cameras. This is based on a recent variational approach by Yezzi and Soatto [21, 22] which proposed a joint region-based image segmentation and simultaneous 3D stereo reconstruction technique. Note that early variational approaches to the 3D reconstruction problem were pioneered by Faugeras et.al. [23] who however also relied on local feature matching.

In our distortion calibration algorithm, we use a white bar object, made from a foam core as shown in Fig.1 on the left. We capture its views before a dark background with the multi-view stereo rig system, a desktop multi-camera system designed for remote multimedia collaboration, developed by HP Labs [24]. The images of the calibration object captured from three of the five cameras in the rig are given in Fig.1. Many desktop multi-camera systems use wide angle and inexpensive cameras which produce severe distortion effects as can be observed in the given images.

Figure 1: Three out of five camera views of the real calibration object shown on the left.

As we will show, with this technique we can also incorporate other parameters of calibration into the same variational framework and get their estimates locally optimally in this sense at one shot.

1.1.2 Color Calibration

Another common problem in multi-view stereo techniques is caused by color miscalibrations between cameras which are variations in camera responses due to different sensor characteristics, ambient conditions like temperature, manufacturing differences, and so on. These result in different color measurements between cameras, and affect the algorithms that depend on these measurements. Camera color calibration refers to the problem of estimating the color calibration parameters of cameras to overcome these unwanted effects. A common approach taken toward this problem is to calibrate each camera independently through comparisons with known colors on a color calibration object/environment [24, 25].

The current color calibration object we use, shown in Fig.2 is a color cube with patches of known colors whose

Figure 2: Photograph of the color calibration object.

images are captured from each camera. Demosaicing coefficients are calculated independently for each camera based upon the absolute colors on the calibration object and the measured color responses of each camera. Slight errors and differences that arise from this independent calibration procedure sometimes lead to noticeable seams or discontinuities in the texture mapping process during the transition of the texture map between neighboring cameras. Our goal is to help even out these discrepancies by devising a *relative inter-camera* color calibration technique in which demosaicing parameters of cameras are calculated jointly in a tightly coupled framework rather than just one camera at a time.

Similar to our approach to lens distortion calibration, the desired coupling of the multiple camera views comes from estimating a common 3D shape, and in addition a common radiance function for the calibration object (in this case, the color cube). We take advantage of the fact that the object shape is known up to location and scale to simplify the problem. Hence, we estimate the pose parameters of the cube, the radiance function on the cube, and the color calibration coefficients for each camera.

A nice feature of the methodology presented in this paper is that it can integrate a whole bunch of small and different problems such as distortion calibration, color calibration into an overall unified system based on the joint segmentation framework, and simultaneously evolve pose, color, distortion, extrinsics, and other parameters as well.

The organization of this paper is as follows. We first present a variant of the Yezzi-Soatto algorithm in which a 3D object is allowed to move with a semi-affine motion model in Section 2. We then present a novel technique for lens distortion calibration in Section 3 and a novel technique for relative inter-camera color calibration in Section 4. Conclusions and discussions are given in Section 5.

2. Evolution Equations of 3D Object Motion Parameters

The Yezzi-Soatto 3D stereo reconstruction model builds a cost on the discrepancy between the reprojection of a model

surface with a radiance $\boldsymbol{f}$ (radiance of background $\boldsymbol{g}$), and the actual measurements from multiple camera views. Let g_i denote the transformation from world coordinates to camera coordinates: $g_i : \vec{X} \longrightarrow \vec{X}_i$, and π denote the perspective transformation from camera frame to the image plane: $\pi : \vec{X}_i = \longrightarrow \vec{x}_i = (x_i = \frac{X_i}{Z_i}, y_i = \frac{Y_i}{Z_i})^T$. On the image plane, the cost functional for the Yezzi-Soatto model can be written as a joint segmentation problem over regions of n camera images I_i:

$$E = \sum_{i=1}^{n} \int_{R_i} [\boldsymbol{f}\,((\pi \circ g_i)^{-1}(\vec{x}_i)) - I_i(\vec{x}_i)]^2 d\vec{x}_i + \int_{R_i^c} [\boldsymbol{g} - I_i]^2 d\vec{x}_i \tag{1}$$

This energy can be lifted back onto surface $\mathbf{S}$:

$$E(S) = \sum_{i=1}^{n} \int_{\mathbf{S}} (\boldsymbol{f}\,(\vec{X}) - I_i(\pi \circ g_i(\vec{X})))^2 - (\boldsymbol{g} - I_i)^2]\, \mathcal{X}_i(\vec{X}) \sigma(\vec{X}) dA \tag{2}$$

where σ is the Jacobian of the change of coordinates from the image plane to the surface, and $\mathcal{X}_i$ is the visibility function of a voxel on the surface. The deformation of the surface $\mathbf{S}$ w.r.t. this energy or data fidelity measure is then obtained by finding the Partial Differential Equation (PDE) which is the gradient descent flow of the energy E. A popular class of numerical techniques known as Level Sets Methods [26], is utilized to evolve the surface $\boldsymbol{S}$ via the evolution of a 3D function $\Psi : \mathbb{R}^3 \longrightarrow \mathbb{R}$. Nevertheless, an update of the level set function is required after each iteration of the associated PDE, and even with more efficient narrow-band schemes [27], there is a considerable amount of computation involved. For our intended applications, in which there is a calibration object whose shape can be roughly known (up to 3 scales and rigidity, which we refer to as semi-affine), rather than deforming the surface of the 3D object, we will evolve its pose and scale parameters instead to obtain more efficient and faster algorithms.

Next, we will derive the equations to update the parameters of the surface motion. Let the original rigid surface be denoted by $\mathbf{S_o}$, then $\mathbf{S} = g(\mathbf{S}_o)$, or $\vec{X} = g(\vec{X}_o) = R_s S \vec{X}_o + T_s$, (where S is a nonuniform scaling matrix) and let λ denote parameters of the motion g. Then the gradient of the energy E w.r.t. λ is given by:

$$\begin{aligned} \frac{\partial E(\lambda)}{\partial \lambda} &= \int_{\mathbf{S}} \sum_i F_i(\vec{X}) < \frac{\partial \vec{X}}{\partial \lambda}, \vec{N} > dA, \\ &= \int_{\mathbf{S_o}} \sum_i F_i(g(\vec{X}_o)) < \frac{\partial (g\vec{X}_o)}{\partial \lambda}, R_s \vec{N}_o > dA_o \end{aligned} \tag{3}$$

where F_i includes data and visibility related terms in (2). Then the gradient term in (3), for translation parameters is,

$$< \frac{\partial (g\vec{X}_o)}{\partial \lambda}, R_s \vec{N}_o > = R_s \vec{N}_o,$$

and for rotation parameters:

$$\begin{aligned} & < \frac{\partial (g\vec{X}_o)}{\partial \lambda}, R_s \vec{N}_o > = \\ = \; & < R_s \begin{bmatrix} 0 & Z_o & -Y_o \\ -Z_o & 0 & X_o \\ Y_o & -Xo & 0 \end{bmatrix}, R_s \vec{N}_o >, \end{aligned} \tag{4}$$

where we utilize exponential coordinates [28], and the gradient term for scaling parameters is given by

$$< \frac{\partial (g\vec{X}_o)}{\partial \lambda}, R_s \vec{N}_o > = < R_s \frac{\partial S}{\partial \lambda} \vec{X}_o, R_s \vec{N}_o > . \tag{5}$$

Note that we can generalize this idea in a straightforward fashion by considering S to be more general than a simple diagonal matrix in order to accomodate a fully affine motion of the surface. We use these evolution equations in updating the pose of the surface $\mathbf{S}$ to estimate its correct placement in the 3D space for the calibration applications presented in Sections 3 and 4.

3. Lens Distortion Calibration

The lens distortion is usually modeled by a function defined from the ideal image plane to the distorted image plane. One approach is to decompose it into two terms: radial and tangential distortion [10]. The radial distortion is a deformation along the radial direction from a center of distortion point to an image point, and the tangential distortion is a deformation in a direction perpendicular to the radial direction, and is negligible in the context of many computer vision applications. To model the radial distortion effects, a commonly used distortion function $D(r)$ is given by $(1 + k_1 r^2 + k_2 r^4 + ...)$ where r is the radius from the center of distortion to a point on the ideal image plane. The principal point is often used as the center for radial distortion [3], which we will also adopt. Below $\vec{\hat{x}_i}$ is the distorted image coordinates, and D is the distort function:

$$\vec{\hat{x}_i} = D\vec{x}_i = (1 + k_1^i r^2 + k_2^i r^4 + ...)\vec{x}_i, \tag{6}$$

$r^2 = (x_i^2 + y_i^2)$, and k_j^i is the j^{th} distortion coefficient for camera i.

The gradient of the energy in (1), defined over the distorted image plane instead of the ideal plane, w.r.t. distortion parameters k_j^i is given by

$$\frac{\partial k_j^i}{\partial t} = \frac{\partial E}{\partial k_j^i} = -\int_{\hat{C}_i} F_i((D \circ \pi \circ g_i)^{-1} \vec{\hat{x}_i}) < \frac{\partial \vec{\hat{x}_i}}{\partial k_j^i}, \vec{\hat{n}_i} > d\hat{s} \tag{7}$$

where $\hat{s}$ is the arclength of the contour $\hat{C}_i$ on the image plane (the distorted or actual observed image coordinates), and note that we assumed a piecewise constant radiance model on the surface and the background to simplify the computations.

We can compute $\frac{\partial \vec{x_i}}{\partial k_j^i}$ as $\frac{\partial \vec{x_i}}{\partial k_1^i} = r^2\vec{x_i}$, $\frac{\partial \vec{x_i}}{\partial k_2^i} = r^4\vec{x_i}, \cdots$ hence $\frac{\partial \vec{x_i}}{\partial k_j^i} = r^{2j}\vec{x_i}$, and

$$< \frac{\partial \vec{x_i}}{\partial k_j^i}, \vec{n_i} > d\hat{s} = < r^{2j}\pi(\vec{X_i}), J\frac{\partial}{\partial s}(D \circ \pi)\vec{X}_i > ds$$
$$= < r^{2j}\pi(\vec{X_i}), JD' \circ \pi' \frac{\partial}{\partial s}\vec{X}_i > ds \quad (8)$$

where J denotes the 2×2 ninety degree rotation matrix, $D' = (1 + k_1^i r^2 + k_2^i r^4 + ...)$, and π' denotes the Jacobian of π. After lifting the integral in Eq.(7) back onto occluding boundary C_i of the surface, and some further manipulations, we get the calibration equation,

$$\frac{\partial k_j^i}{\partial t} = \int_{C_i} F_i \frac{r^{2j} D' ||\vec{X}_i||}{Z_i^3} < \vec{N}_i, \begin{pmatrix} X_i \\ Y_i \\ 0 \end{pmatrix} > ds \quad (9)$$

for the lens distortion parameter k_j^i. Note that the distortion calibration method we propose can handle different models of distortion by changing the D function, and related derivatives above.

Using Several Poses of the Object: When camera views from multiple poses of the object are available, we can take advantage of the existence of variously distorted views in calibrating the lens distortion. In the first phase, we estimate both pose and distortion coefficients from separate experiments. To simplify the explanation, let us assume that we want to solve for only one distortion coefficient k_i for each camera i. Once we obtain rough estimates for the object pose and distortion coefficients k_i, we can fuse a "common distortion" $\tilde{k_i}$ from these separate experiments for each camera i and then jointly evolve $\tilde{k_i}$'s as follows:

$$\frac{\partial \tilde{k_i}}{\partial t} = \sum_{m=1}^{M \text{poses}} \int_{C_m} F_{i,m} \frac{r^{2j} D' ||\vec{X}_{i,m}||}{Z_{i,m}^3} < \vec{N}_{i,m}, \begin{pmatrix} X_{i,m} \\ Y_{i,m} \\ 0 \end{pmatrix} > ds \quad (10)$$

At the same time, we evolve the pose parameters of separate poses of the object, only difference being the incorporation of the new "common distortion" in the equations. For instance, we evolve any of them for a given pose as:

$$\frac{\partial \lambda}{\partial t} = -\sum_i \int_{S_o} F_i(g(\vec{X}_o)) < \frac{\partial (g\vec{X}_o)}{\partial \lambda}, R_s \vec{N}_o > dA_o$$

where F_i includes computation of $I_i(D \cdot \pi \cdot g_i\ (g(\vec{X}_o)))$ with the new common distortion coefficients $\tilde{k_i}$ in the multiplying distortion factor D.

Experimental Results: For our calibration algorithm, we initialize a surface model of the real calibration object which is shown from several vantage points in Fig.3. After initializing the surface, the first phase of our algorithm is to evolve its pose parameters to position the 3D object model roughly in the correct location in 3D space. For the experiments presented here, we have used three different poses of the calibration object, but we can increase the number of poses used in the process. Example evolutions of the pose parameters are shown in Fig.4, for three different pose captures of the calibration object in each column (showing only one camera view for each pose). The distortion coefficients are also evolved at a slower pace. That is, the time step used in the associated PDE is small in the first phase. In the experiments, the distortion function D in (6) with one distortion coefficient k_1 for each camera is used.

After the separate evolutions for each of the poses have converged, common initial distortion coefficients are computed as the average of the results from phase 1. In the second phase of the algorithm, we evolve the distortion coefficients for each camera again separately but summed over different poses. We show sample views of pose 1, 2, and 3 in row 1 of Fig.s 5-6-7. As the distortion coefficients converge to true values, the reprojection of the surfaces onto the distorted views results in a better match to the image data and continues to minimize the overall energy. Such images with reprojections are shown on the second row of Figures 5-6-7. The undistorted views shown as well on the third row. The straightening effect of this operation on the curved lines can be clearly observed in these images.

Figure 3: Initialized surface model shown from three different vantage points.

4. Color Calibration

For color calibration purpose the differences in absolute colors measured in the response of each camera are modeled by a simple multiplicative factor in each of color RGB channel measurements and an additive offset parameter.

The first variation of our energy functional E using this model leads to gradient descent flows:

$$\frac{\partial E}{\partial \alpha_{i,j}} = -\int_{R_i} [\boldsymbol{f} - (\alpha_{i,j} I_{i,j} + \beta_{i,j})] I_{i,j} d\vec{x} - \int_{R_i^c} [\boldsymbol{g} - (\alpha_{i,j} I_{i,j} + \beta_{i,j})] I_{i,j} d\vec{x},$$
$$\frac{\partial E}{\partial \beta_{i,j}} = -\int_{R_i} [\boldsymbol{f} - (\alpha_{i,j} I_{i,j} + \beta_{i,j})] d\vec{x} - \int_{R_i^c} [\boldsymbol{g} - (\alpha_{i,j} I_{i,j} + \beta_{i,j})] d\vec{x}$$

for the color calibration parameters $\alpha_{i,j}$ and $\beta_{i,j}$ for each camera i, and $j \in \{R, G, B\}$, where $I_{i,j} \in \{R_i, G_i, B_i\}$. Note that one can extend this framework to RGGB images in a straightforward fashion.

In our experiments, we utilized white noise additive offsets and multiplicative scaling coefficients to perturb the measured images, thereby exaggerating the effect of color miscalibrations. On a synthetically created grayscale example in Fig.8, where the correct geometry and radiance function are known, we show such miscalibration effects on the original views, and views during the evolution of α_i's and β_i's, and views after these parameters have converged. In addition, in Fig.9, the curves depict the true α and β values for all nine camera views, and the convergence of the estimated parameters towards the real values.

As for a real calibration experiment, we captured images of the calibration object from five cameras, three of which are shown in Fig.10[1]. Similarly, the same views under different conditions are shown on the second row of the same figure where the first picture is darker than the original one, second and third pictures are lighter than the original ones. We initialize a model cube, and a radiance function on the cube, as shown in bottom two rows of Fig.10, the initial and the final values respectively. The third row from the top shows views after the evolution of color calibration coefficients are completed. The fourth row shows the projections of the model surface onto the views. Similarly, in Fig.11, another experiment depicting the views before and after the color calibration is presented. It can be visually assessed that color responses of the cameras have evolved towards their correct ranges.

5. Conclusions and Discussions

In this paper, we applied stereoscopic variational ideas to various camera calibration problems. We have presented new multi-view stereo techniques to: evolve pose parameters of a 3D model object to take advantage of the known shape of calibration object and reduce computational load; evolve distortion and color calibration camera parameters.

As for the distortion calibration method, utilizing more poses, hence many more camera images of the calibration object, and several distortion coefficients rather than one in the model selected, will lead to improvements in the estimation. We can also utilize more general/complicated distortion models than the simple polynomial D function.

The presented methods eliminate the need for search of image edges, point correspondences from images, which can be very sensitive to pixel-level noise whereas our approach being based on image regions for comparisons, is not as much sensitive to noise as the former. Another advantage of this framework is that it easily accomodates additional data. In the more classical approaches to stereo, bringing in more data, or adding more images to the algorithm might not help all the time, that is if something goes wrong in the independent segmentation phase of even one image, it destroys the whole process of reconstructions and geometry. On the other hand, adding more data to this joint segmentation framework, will only improve robustness, provide more tolerance towards errors and nothing else. Last but not the least, this variational framework enables one to be able to incorporate other camera calibration parameters into the joint energy functional for all cameras, and find locally optimal solutions.

[1]Color image experiments can be viewed at the website: http://users.ece.gatech.edu/ $\sim$ ayezzi

References

[1] I. Sobel, "On calibrating computer controlled cameras for perceiving 3d scenes," *AI*, vol. 5, pp. 185–198, 1974.

[2] R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *J. Robotics and Automation*, vol. 3, pp. 323–44, 1987.

[3] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.

[4] B. Triggs, "Camera pose and calibration from 4 or 5 known 3d points," in *Proc. of CVPR*, 1999.

[5] P. Gurdjos and P. Sturm, "Methods and geometry for plane-based self-calibration," in *Proc. of the CVPR*, 2003.

[6] O.D. Faugeras, Q.T. Luong, and S.J. Maybank, "Camera self-calibration: Theory and experiments," in *Proc. of ECCV*, 1992.

[7] Y. Seo, A. Heyden, and R. Cipolla, "A linear iterative method for auto-calibration using the dac equation," in *Proc. of CVPR*, 2001.

[8] J. Oliensis, "Fast and accurate self calibration," in *Proc. of ICCV*, 1999.

[9] J. Heikkila and O.Silven, "A four-step camera calibration procedure with implicit image correction," in *CVPR*, 1997.

[10] F. Devernay and O. Faugeras, "Straight lines have to be straight," *Mach. Vis. and Appl.*, vol. 13, 2001.

[11] J. Weng, P. Cohen, and M. Herniou, "Camera calibration with distortion models and accuracy evaluation," *IEEE PAMI*, vol. 14, pp. 965–980, 1992.

[12] Z. Zhang, "A flexible new technique for camera calibration," Tech. Rep., Microsoft Research, MSR-TR-98-71, 1998.

[13] D.C. Brown, "Decentering distortion of lenses," *Photogrammetric Engineering*, vol. 32, 1966.

[14] M. T. El-Melegy and Aly A. Farag, "Nonmetric lens distortion calibration: Closed-form solutions, robust estimation and model selection," in *Proc. of ICCV*, 2003.

[15] S.B. Kang, "Semiautomatic methods for recovering radial distortion parameters from a single image," Tech. Rep., CRL Tech. Report CRL 97/3, 1997.

[16] R. Swaminathan and S.K. Nayar, "Nonmetric calibration of wide-angle lenses," in *Proc. of CVPR*, 1999.

[17] G.P. Stein, "Lens distortion calibration using point correspondences," in *Proc. of CVPR*, 1997.

[18] A.W. Fitzgibbon, "Simultaneous linear estimation of multiple view geometry and lens distortion," in *CVPR*, 2001.

[19] F. Du and M. Brady, "Self-calibration of the intrinsic parameters of cameras for active vision systems," in *Proc. of CVPR*, 1993.

[20] H.S. Sawhney and R. Kumar, "True multi-image alignment and its application to mosaicing and lens distortion correction," in *Proc. of CVPR*, 1997.

[21] A. Yezzi and S. Soatto, "Stereoscopic segmentation," *Int. J. Computer Vision*, vol. 53, pp. 31–43, 2003.

[22] A. Yezzi and S. Soatto, "Structure from motion for scenes without features," in *CVPR*, 2003.

[23] O. Faugeras and R. Keriven, "Variational principles, surface evolution pdes, level set methods and the stereo problem," Tech. Rep., INRIA, 1996.

[24] H.H. Baker, D. Tanguay, I. Sobel, D. Gelb, M.E. Goss, B.W. Culbertson, and T. Malzbender, "The coliseum immersive teleconferencing system," Tech. Rep., HP Labs, HPL-2002-351, 2002.

[25] E. Marszalec, "On-line color camera calibration," in *Proc. of ICPR*, 1994.

[26] S. Osher and J.A. Sethian, "Fronts propagating with curvature dependent speed," *JCP*, vol. 49, 1988.

[27] D. Adalsteinsson and J.A. Sethian, "A fast level set method for propagating interfaces," *JCP*, vol. 118, 1995.

[28] R.M. Murray, Z. Li, and S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, 1994.

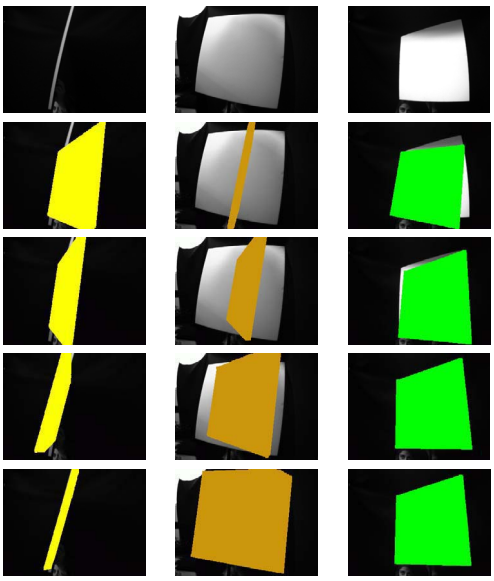

Figure 4: Column 1: Pose1. Row 1: one out of five camera image shown, Row2: with projection of initialized surface (orange mask), Rows 3-4: during evolution of the pose parameters of the surface, Row 5: with converged pose parameters. Columns 2-3: same as column 1 for poses 2 and 3, respectively.

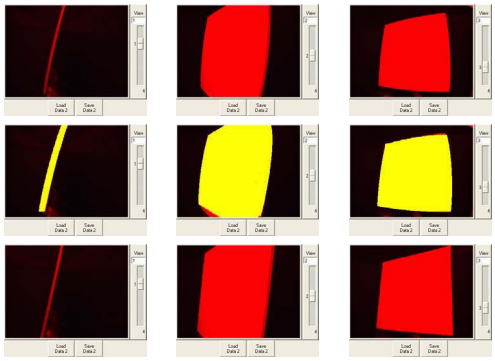

Figure 5: Pose 1. Row 1: Three out of five captured views. Row 2: Projected surface after distortion parameters have converged. Row 3: Undistorted with the obtained distortion coefficients.

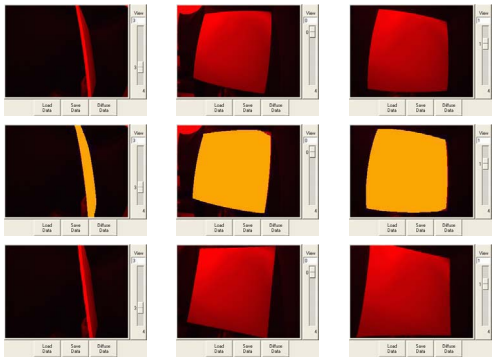

Figure 6: Pose 2. Row 1: Three out of five captured views. Row 2: Projected surface after distortion parameters have converged. Row 3: Undistorted with the obtained distortion coefficients.

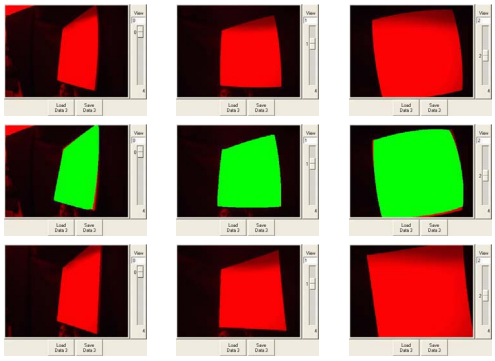

Figure 7: Pose 3. Row 1: Three out of five captured views. Row 2: Projected surface after distortion parameters have converged. Row 3: Undistorted with the obtained distortion coefficients.

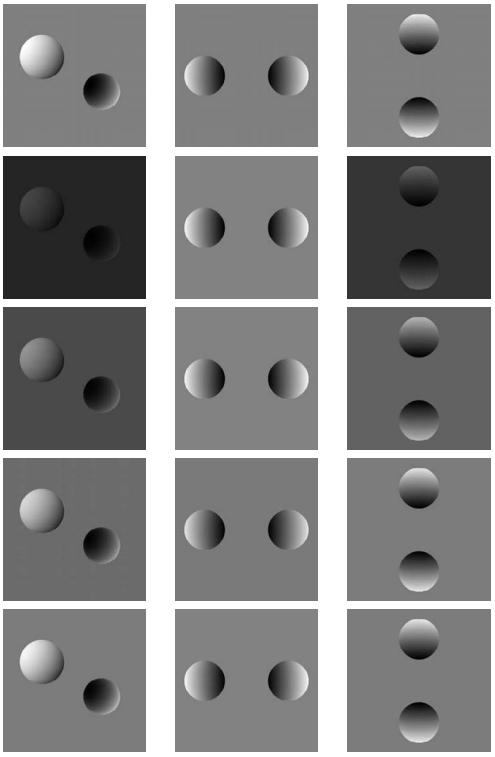

Figure 8: Row 1: Three original views. Row 2: The same three different after deliberate simulated miscalibration of the greyscales. The same three views while evolving the calibration parameters: Rows 3-4 intermediate stages, Row 5: The views after evolution of the calibration parameters has completed.

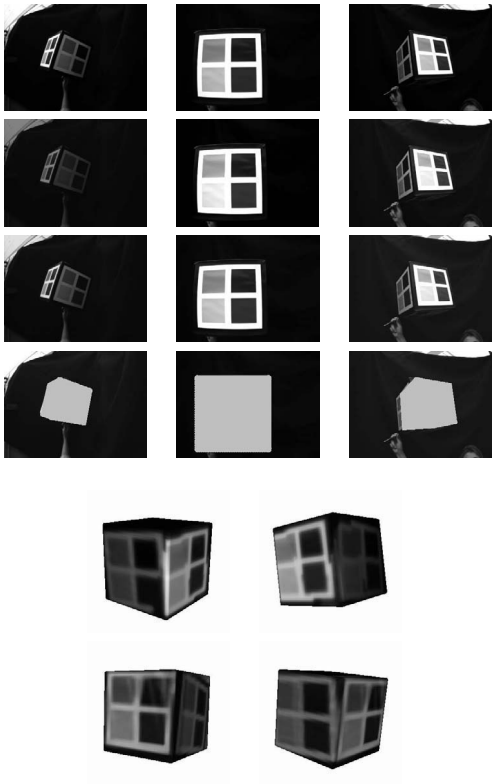

Figure 10: Row 1: Three original views. Row 2: The same three different after deliberate simulated miscalibration of the greyscales. Row 3: The views after evolution of the calibration parameters has completed. Row 4: The projections of the model surface onto the same views. Rows 5-6: Radiance reconstructed on the cube from camera views shown at different vantage points.

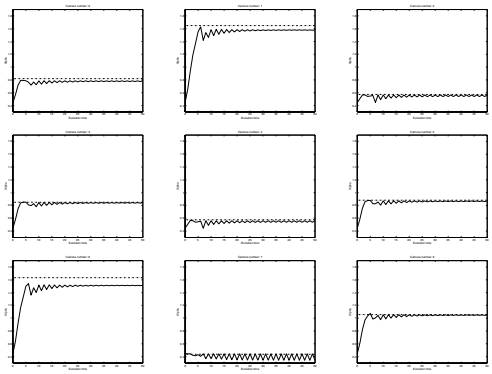

Figure 9: Evolution of the parameter α for different views. (True α value shown as a dotted line)

Figure 11: Row 1: Three original views. Row 2: The same three different after deliberate simulated miscalibration of the greyscales. Row 3: The views after evolution of the calibration parameters has completed.