An Inexpensive, Automatic and Accurate Camera Calibration Method

Keith Forbes
Digital Image Processing Group
Department of Electrical Engineering
University of Cape Town
Private Bag, Rondebosch, 7701
kforbes@dip.ee.uct.ac.za

Anthon Voigt
Machine Intelligence Group
DebTech
De Beers Consolidated Mines
PO Box 82851, Southdale, 2135
anthon.voigt@debeersgroup.com

Ndimi Bodika
Machine Intelligence Group
DebTech
De Beers Consolidated Mines
PO Box 82851, Southdale, 2135
ndimi.bodika@debeersgroup.com

Abstract

A calibration procedure for accurately determining the pose and internal parameters of several cameras is described. Multiple simultaneously-captured sets of images of a calibration object in different poses are used by the calibration procedure. Coded target patterns, which serve as control points, are distributed over the surface of the calibration object. The observed positions of these targets within the images can be automatically determined by means of code band patterns. The positions of the targets across the multiple images are then used to infer the camera parameters, as well as the 3D geometrical structure of the targets on the calibration object (thus avoiding the expense of a calibration object with accurately known 3D structure). Results for a three-camera system show RMS (root-mean-square) deviations of less than five microns of the inferred positions of 54 control points, distributed on the surface of a 50 mm cube, from their expected positions on a flat surface. The RMS difference between the positions of 1423 observed control points and the positions predicted by a 330 parameter model of the camera system and calibration object was 0.09 pixels.

1 Introduction

Most commonly used calibration procedures described in the computer vision literature rely on a calibration object with control points whose 3D coordinates are known with a high degree of accuracy [8, 5]. Control points are typically positioned on two or three faces of a cube-shaped frame so that they are well distributed in all three dimensions. Since the 3D coordinates of the control points are known, it is possible to infer camera pose (relative to the calibration object's reference frame) and internal parameters from the 2D positions of the imaged control points measured from within a single image. For accurate results (i.e. results in which the actual locations of imaged points are very close to the locations predicted by the inferred camera parameters), the 3D positions of the control points must be known accurately. Having the 3D positions measured accurately by a metrology laboratory is a costly procedure [10].

Recently, methods that rely on multiple images of a planar calibration object in different poses have become popular. The popularity of these planar methods is mainly because it is easy and inexpensive to obtain a relatively accurate planar calibration object. This can be done by using a laser printer to print a pattern and then affixing it to a flat object such as a piece of glass. By increasing the number of images of the planar calibration object in different poses, it is also possible to reduce the effect of image noise on the accuracy of the inferred camera parameters.

Freeware implementations of both single image calibration using non-coplanar control points and multiple image calibration using coplanar control points are available*. The planar calibration method provides an estimate of the camera pose with respect to each calibration object pose. In order to determine the relative poses of two or more cameras, each pose of the calibration object must be viewed by all of the cameras. Each calibration object pose provides an estimate of the relative poses of the cameras. The planar calibration method is therefore useful in providing accurate estimates of the internal parameters of a camera (assuming that the 3D coordinates of the control points are accurately known), but does not provide a means for using all available observations (2D locations of imaged-control points) to estimate relative camera poses using all observations. The relative camera pose that is used to model the imaging process must either be selected from one of the relative pose estimates, or all of the pose estimates may be combined using a singular value decomposition approach. The pose and internal parameter estimates may also be used as a starting point for an optimisation procedure that minimises the reprojection residual across all observations. It is also possible to refine the estimates of the 3D coordinates of the control points (which are limited to the accuracy of the laser printer used to print the pattern, and to a lesser extent by the flatness of the pattern on the glass backing) by including the 3D coordinates as parameters in the minimisation process. This refinement is particularly important when calibrating camera systems for measuring small objects, since the magnitude of the errors associated with the 3D coordinates of the control points (due to inaccuracies in the laser printing process) may be significant with respect to the size of the measured objects. A major drawback of the planar calibration system that is followed by a simultaneous optimisation of all parameters, is that all of the control points must be visible in all of the images. This is often difficult to arrange for a multi-camera setup.

The calibration procedure described in this paper makes use of multiple poses of a polyhedral calibration object with coded targets as control points on the object's faces. The coded targets allow the positions of the control points in each image to be automatically measured to sub-pixel accuracy, i.e. human intervention is not required in order to identify corresponding control points across multiple images. The procedure uses the approximately known 3D structure of the calibration object and the 2D positions of control points across multiple images to form an initial approximate solution to the calibration parameters. This is done by forming an estimate of the focal length of each camera using Tsai's method and using singular value decomposition to combine pose estimates obtained using a recently developed globally convergent pose estimation algorithm. The initial approximate solution is then used as a starting point for a bundle adjustment procedure that

*Reg Willson's C implementation of Tsai's algorithm [8] (see `http://www.cs.cmu.edu/afs/cs.cmu.edu/user/rgw/www/TsaiCode.html`) and Jean-Yves Bouguet's MATLAB implementation based on Zhang's algorithms [11] (see `http://www.vision.caltech.edu/bouguetj/calib_doc/index.html`) are both popular. Bouguet's implementation has also been ported to C and incorporated into the Intel's OpenCV library (see `http://sourceforge.net/projects/opencvlibrary/`).

computes calibration parameters that are jointly optimal across all observed imaged control points. The calibration parameters consist of accurate estimates of camera focal lengths, principal points and first order radial lens distortion coefficients as well as camera poses and 3D coordinates of control points.

The calibration procedure thus provides accurate estimates of internal and pose parameters without requiring the accurate prior knowledge of the 3D structure of the calibration object that is needed for calibration procedures such as Tsai's. This is achieved by making use of multiple poses of the calibration object so that the 3D structure can be inferred from the positions of corresponding control points across multiple images. The effects of measurement error of the 2D positions of the imaged control points on the computed calibration parameters can be reduced by increasing the number of poses of the calibration object that is used. Unlike the planar calibration methods, all control points need not be visible in all views.

As a side-effect of the calibration procedure, the 3D structure of the calibration object is accurately determined. The calibration procedure could therefore be useful for purposes other than determining the camera parameters; the method may be used to accurately determine the 3D coordinates of control points on a calibration object. A single camera may subsequently be rapidly calibrated with a single image using the calibration object and a method such as Tsai's that relies on accurate prior knowledge of the 3D structure of the calibration object.

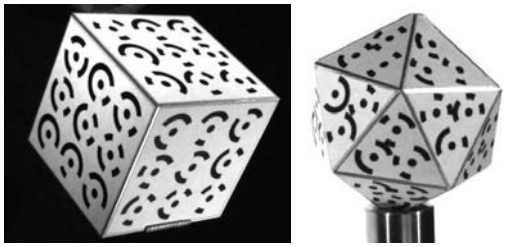
The design of the coded targets and the image processing procedures used to locate and identify the imaged control points is described in Section 2 of this paper. Section 3 describes a routine that provides additional robustness to the entire calibration procedure by rejecting identified control points that may have been misclassified by the target location and identification routine. This is done by rejecting targets that do not conform to the approximately known 3D structure of the calibration object. Section 4 describes how the calibration parameters are estimated by forming an initial estimate based on both the observed 2D positions of the imaged control points and the approximately known 3D structure of the calibration object. It is also described how this estimate is refined using a bundle adjustment procedure that takes the circular shape of the targets into account. This is followed by an explanation of how scale is enforced by means of Procrustes analysis. Section 5 describes an experiment that was performed to measure the performance of the calibration procedure. Section 6 provides a summary of the paper and conclusions.

2 Coded Targets

In order for the calibration process to be automated, imaged control points must be automatically located and identified within images. These control points are the centres of coded targets, that are patterns on the faces of a polyhedral calibration object. Two calibration objects are shown in Figure 1. In this section, the design of the circular coded targets and the image processing algorithms used to locate and identify images of coded targets are described.

2.1 Target Design

Ahn *et al.* [1] provide a short overview of ten different coded target pattern designs that have been used for automating 3D measurement processes. They also describe their own coded target design that is based on a central circular target surrounded by a pattern of smaller dots that encode the identity of the circular target. The coded target design described here is similar to some of the existing designs in that each target consists of a circle surrounded by a code band. However, unlike the designs described in the literature, no start-, stop-, or parity-bits are used. This is because noise is likely to be distributed relatively evenly over the region surrounding the imaged target, rather than being localised.

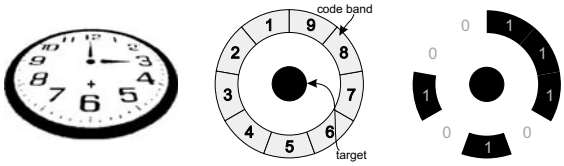


(a) (b)

Figure 1: Two calibration objects with coded targets: (a) a cube with 54 targets and 9-bit code bands, (b) an icosahedron with 60 targets and 10-bit code bands.

A target must be designed so that the positions of images of a 3D control point can be measured across multiple images to subpixel accuracy. The centre of the target serves as the control point.

Circular targets were used, because circles are imaged as ellipses: ellipses can be automatically detected with relative ease from within images. The centre of an image of a circle is a very good approximation to the image of the centre of the circle [2]. This is the case unless the size of the circle is large with respect to the distance to the camera (as is shown in Figure 2(a)). Each



(a) (b) (c)

Figure 2: Diagrams showing (a) a perspective projection of a clock, illustrating the difference between the centre of an imaged-circle (plus sign) and the image of a centre of a circle (black dot); (b) the layout of the target and code band pattern; and (c) an example of the code 001010111_2 with the bit values overlaid in grey.

circular target (black dot) on the calibration object is surrounded by a unique code band pattern that is used to identify the target. Figure 2(b) shows the structure of a coded target. The central dot is surrounded by a code band with bit positions at equally spaced angular intervals. Each of the bit positions can be either black or left empty (white).

Each bit pattern has a number associated with it. To decode the pattern, the binary code is read anticlockwise. Each bit is considered to be the first bit in turn. This means that for a k-bit code, there are k binary numbers to be considered. The number corresponding to the code pattern is the lowest of these k numbers. For instance, the 9-bit code shown in Figure 2(c) corresponds to the binary sequences 001010111, 010101110, 101011100, 010111001, 101110010, 011100101, 111001010, 110010101, 100101011 (black or inked regions represent '1's and white or empty regions represent '0's). Of these nine binary numbers, 001010111, has the lowest value ($001010111_2 = 87_{10}$), so this code pattern is labelled as number 87. Note that not all positive integers correspond to code patterns. The minimum of all possible rotationally shifted binary codes is simply used as a number to identify the code pattern.

2.2 Identifying and Locating Targets

The identification process aims to identify connected regions of pixels that represent the boundaries of the elliptical imaged targets. This is done by identifying edge pixels within the image and then classifying connected regions of edge pixels as either target or non-target regions.

Firstly, Canny edge detection is applied to the image. Edge pixels are then labelled in groups of connected components, each to be classified as either target or non-target groups.

Candidate target regions (connected components) are considered for rejection with a cascade-type classifier that measures increasingly complex features. Using this method, obvious non-targets are quickly rejected without the computational expense of measuring more complex features.

Features of the candidate regions are measured in the following order:

1. **Number of pixels**: If the number of pixels in a region is too high or too low, the region is rejected, and no further features are measured.
2. **Distance to image boundary**: If the region is too close to the boundary of the image, the code band will not be entirely visible. Regions that are close to the image boundary are rejected.
3. **Euler number**: Regions must have a single hole.
4. **Fit of best fit ellipse**: An ellipse is fitted to the coordinates of the pixels in the candidate region. If the sum of squared distances from the coordinates of the pixels in the candidate regions to the fitted ellipse is too great, then the region is rejected.
5. **Contrast**: The foreground region within the fitted ellipse must be sufficiently darker than the surrounding background.
6. **Fit of best fit code**: The error in the fit of the best fit bit pattern must be sufficiently low for the region to be classified as a target.

Ellipse fitting is performed using the method of Fitzgibbon *et al.* [4]. The method is computationally efficient, and unlike other available algebraic methods, an ellipse is fitted, rather than a general conic.

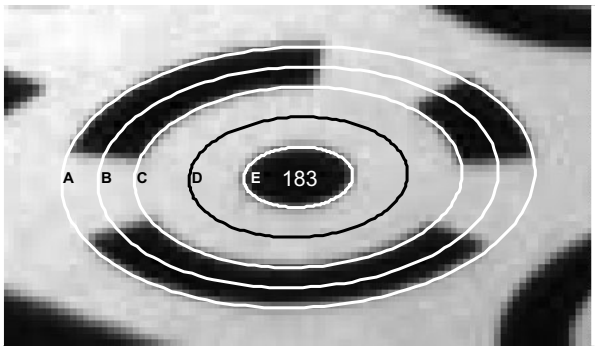


Figure 3: Coded target 183 ($010110111_2 = 183_{10}$) with fitted ellipse (labelled E), ellipse on which background pixel intensity is observed (labelled D), the inner and outer boundaries of the code band (labelled C and A), and the ellipse on which the code band pixel intensities are observed (labelled B).

Figure 3 shows an example of an ellipse (labelled E) that has been fitted to the edge pixels of the image of a circular coded target. Since the geometry of the coded target is known, ellipses that define the inner and outer boundaries of the code band (labelled C and A), the centre of the code band (labelled B) and midway between the code band and target (labelled D), can be computed. This is because the geometry of the region surrounding the imaged dot is an approximately affine transformation of the physical planar region surrounding the dot on the surface of the calibration object. The approximation holds because the depth of the region is small with respect to the distance to the camera centre.

The median value of all pixels that lie on the ellipse midway between the code band and target is used to define the background pixel intensity value. The median value of all pixels in the region that is bounded by the target edge pixels is used to define the foreground pixel intensity value. These values are used to determine the contrast (difference between foreground and background values) and for determining the the best fit code on the code band.

The corresponding code for each candidate target is computed by minimising a function of one variable. The intensity values of all pixels lying on the ellipse that runs through the centre of the code band are determined. The coordinates of these pixels are normalised so that each intensity value corresponds to a position on a unit circle, rather than on an ellipse. Starting at an angle of θ, the unit circle is divided into k bit segments (for a k-bit code), each with an angular extent of $360°/k$. The mean intensity value corresponding to each bit segment is used to determine whether the bit segment is a‘1’ or a ‘0’. The mean of foreground and background pixel values is used as a threshold. The error associated with a given value of θ is the sum of squared differences between the observed intensity values and the background value (in the case of bit segments designated as ‘0’) or the foreground value (in the case of bit segments designated as ‘1’). This error is minimised (in order to determine the value of θ to use) using the MATLAB `fminbnd` function which is based on Golden Section search and parabolic interpolation [3].

Rather than using the centre of the fitted ellipse as the centre of the imaged target, the intensity weighted centroid is used. This method is typically the more accurate method of determining the centre of the imaged target [7]. Since foreground pixels are black, the inverse of the original grey-scale image must be used. Pixels corresponding to edge pixels and pixels surrounded by the edge pixels specify the regions to consider in this computation.

3 Rejecting False Positives

The known dimensions of the polyhedral calibration object (e.g. 50 mm cube or 20 mm icosahedron) can be used together with the known 2D positions of the control points on each face to determine the approximate 3D structure of the control points. Using this approximately known 3D structure, image points that have been incorrectly labelled as coded targets can be eliminated. This is achieved using the RANSAC (*ran*dom *sa*mple *c*onsensus) paradigm.

A typical image of a calibration object may contain 25 targets that have been identified and located within the image. These targets have usually been correctly identified by the routines described in the previous section, but occasionally a target is incorrectly labelled, or a non-target region is falsely classified as a target. (Targets may also be missed by the location and identification routines so that a target is misclassified as a non-target: the only consequence of this is that there are effectively fewer visible targets in the image for the subsequent calibration routines.)

The RANSAC procedure selects a random sample of seven of the target points within an image and uses their approximately known geometrical structure (3D coordinates) and observed image locations (2D coordinates) to estimate the positions of the projec-

tions of the remaining world points using Tsai's method [8]. The distance between the projected world points and the corresponding observed image coordinates is computed, and points are labelled as outliers if the distance is greater than some empirically determined threshold (say 5 pixels).

The procedure is repeated many times (say 1000 times) with a different random subset of seven points being chosen on each occasion. Of all the random selections, the camera parameters that result in the smallest number of outliers are selected and are used to identify targets to be eliminated.

4 Parameter Estimation

A vector of calibration parameters consisting of the camera internal parameters (focal lengths, principal points and a first order radial lens distortion parameters), camera poses, calibration object poses and 3D control point positions is used to store all of the parameters for the model of the imaging process. The vector $\mathbf{v}$ is given by

$$\begin{aligned}\mathbf{v} = (\ & f_1, u_{0,1}, v_{0,1}, \kappa_{1,1},\ f_2, u_{0,2}, v_{0,2}, \kappa_{1,2}, \ldots,\ f_n, u_{0,n}, v_{0,n}, \kappa_{n,1},\\ & t_{\text{cam},x,1}, t_{\text{cam},y,1}, t_{\text{cam},z,1}, \alpha_{\text{cam},1}, \beta_{\text{cam},1}, \gamma_{\text{cam},1}, \ldots, t_{\text{cam},x,n},\\ & t_{\text{cam},y,n}, t_{\text{cam},z,n}, \alpha_{\text{cam},n}, \beta_{\text{cam},n}, \gamma_{\text{cam},n}, t_{\text{obj},x,2}, t_{\text{obj},y,2},\\ & t_{\text{obj},z,2}, \alpha_{\text{obj},2}, \beta_{\text{obj},2}, \gamma_{\text{obj},2}, \ldots, t_{\text{obj},x,m}, t_{\text{obj},y,m}, t_{\text{obj},z,m},\\ & \alpha_{\text{obj},m}, \beta_{\text{obj},m}, \gamma_{\text{obj},m}, X_1, Y_1, Z_1, X_2, Y_2, Z_2, \ldots, X_r, Y_r, Z_r)^T\end{aligned}$$

where f, u_0, v_0 and κ_1 describe the internal parameters of cameras 1 to n; $t_{\text{cam},x}$, $t_{\text{cam},y}$, $t_{\text{cam},z}$, α_{cam} (yaw angle), β_{cam} (pitch angle) and γ_{cam} (roll angle) describe the poses of cameras 1 to n; $t_{\text{obj},x}$, $t_{\text{obj},y}$, $t_{\text{obj},z}$, α_{obj}, β_{obj} and γ_{obj} describe calibration object poses 2 to m (the first calibration object pose defines the global reference frame, so no parameters need be stored); and X, Y, and Z describe the 3D coordinates of the control points 1 to r in the reference frame of the first calibration object pose.

A vector $\mathbf{v}$ must be found, so that the sum of squared differences between the imaged control points predicted by the model defined by $\mathbf{v}$ and the observed positions of the imaged control points across all images is minimised. Determination of the starting point $\mathbf{v}_0$ is crucial; without a suitable starting point, the minimisation routine may converge on a local minimum that is far from the global minimum. A well-chosen starting point also reduces the running time of the minimisation routine.

4.1 Determining a Starting Point

In order to form a starting point which can later be refined by an optimisation routine, estimates of the camera poses and internal parameters must be made. These estimates are made by assuming that the approximate 3D structure of the control points is known.

Provided that sufficient control points are visible, Tsai's method, or the DLT (direct linear transform) could be used to estimate the camera internal parameters and camera pose with respect to the calibration object. For an n-camera system in which m calibration object poses are considered, this approach would result in m estimates for each of the internal camera parameters, and nm relative poses from which multiple camera and calibration object poses could be derived.

In principle, it would be possible to use only the n images corresponding to the first calibration object pose to estimate the internal parameters of the n cameras, and the m images corresponding to the first camera to estimate the poses of the calibration object with respect to its first pose. However, this approach only makes use of a very small proportion of the available information and may produce a starting point that is too far from the optimal solution for convergence to occur.

Better methods combine the parameter estimates from the different images to obtain parameter estimates with a smaller degree of uncertainty than any of the individual parameter estimates.

There is far greater uncertainty associated with estimating the Z-coordinate (depth) of the control points in a certain image than the X- and Y-coordinates if the internal parameters of the camera are unknown. This is because changes in both the Z-coordinate and the focal length of the camera affect the scale of the object within the image. The only means for distinguishing the effects of the Z-coordinate from those of the focal length is the perspective distortion within the image. Often the effects of perspective distortion are small and are corrupted by noise, so the estimated f and Z values deviate greatly from their true values, whereas the ratio f/Z is relatively accurate. To prevent large deviations in the Z values from inducing a large degree of disagreement between pose estimates computed from different images, the focal length is kept fixed in estimating pose. By enforcing the focal length values, a better estimate of pose can be obtained than by using the pose parameters estimated by Tsai's method. Tsai's method is applied to each of the images that contain a sufficient number of visible control points. This means that up to m estimates are available for each of the n focal lengths. The median of the focal length estimates is used as the starting point focal length value for each camera.

The pose estimation algorithm of Lu *et al.* [6] is used to estimate the object poses. Firstly, the internal camera parameters are used to determine the normalised image coordinates of the observed control points. The normalised image coordinates are the X- and Y-coordinates of the position of corresponding rays in the $Z = 1$ plane. The radial distortion coefficients and principal points corresponding to the median focal length values are used to undistort the image coordinates. The pose estimation algorithm is then applied to the observed control points corresponding to each image and, in conjunction with the knowledge of the approximate 3D structure of the control points, the pose of the calibration object (with respect to the pose of the camera) is estimated.

There are up to n estimates of the relative pose between the first object pose (which is used as a global reference frame) and each of the remaining $m-1$ object poses. To obtain a single pose estimate from the available estimates, the mean of the available translation vectors is used. The rotation component of the single pose estimate was determined by forming the singular value decomposition of the available rotation matrices:

$$\frac{1}{n}\sum_{i=1}^{n} \mathsf{R}_i = \mathsf{U}\Sigma\mathsf{V}^T \tag{1}$$

The average rotation matrix $\mathsf{R}_{\text{average}}$ was then formed by substituting the matrix Σ with the 3×3 identity matrix $\mathbf{I}$:

$$\mathsf{R}_{\text{average}} = \mathsf{U}\mathbf{I}\mathsf{V}^T \tag{2}$$

This ensures that $\mathsf{R}_{\text{average}}$ is orthonormal.

With an estimate of each calibration object pose with respect to the first object pose, it is possible to obtain up to m estimates of the pose of each camera. These pose estimates are combined to obtain a single pose estimate for each of the n cameras using the method described above.

A vector $\mathbf{v}_0$ containing starting values for all of the calibration parameters is formed. The median focal lengths and corresponding principal point and distortion parameters are used for the internal parameters; translation vectors and Euler angles describing

pose are extracted from the pose matrix estimates for both the camera and the calibration object poses; and the approximately known 3D coordinates of the control points are used for the 3D structure of the control points on the calibration object.

At this stage, the RMS reprojection residuals are computed for the sets of control points corresponding to each calibration object pose. This is done using the starting values of the parameters. If any of the RMS reprojection residuals are above an empirically determined threshold, then the observations corresponding to the object pose are removed from the optimisation procedure. This step provides additional robustness to the calibration process by preventing a small number of poses with poor parameter estimates from affecting the convergence of the remaining parameters.

4.2 Bundle Adjustment

The Levenberg-Marquardt method is used to determine the jointly optimal calibration parameters by minimising the sum of squared differences between the observed image coordinates and those predicted by the model of the imaging process. The vector $\mathbf{v}_0$ is used as a starting point.

In order to obtain accurate results, the imaged centre of the circular target is not assumed to be the same as the centre of the imaged circular target (see Figure 2(a)). Rather, the observations are compared with the predictions of a modelling process in which circles are projected onto the image planes and the centres of the resultant ellipses are used. The undistorted sensor coordinates of the ellipse centres $(x_c, y_c)^T$ are determined using:

$$\lambda \begin{pmatrix} x_c \\ y_c \\ 1 \end{pmatrix} = \mathsf{FHQ}^{-1}\mathsf{H}^T \mathbf{f_3} \tag{3}$$

where F is the matrix to convert world coordinates to undistorted sensor coordinates,

$$\mathsf{H} = \begin{pmatrix} \mathbf{h}_1 & \mathbf{h}_2 & P_0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4}$$

where $\mathbf{h}_1$ and $\mathbf{h}_2$ and are two orthogonal 3D vectors that span the plane on which the circle lies and P_0 is the centre of the circle,

$$\mathsf{Q} = \begin{pmatrix} -1/r^2 & 0 & 0 \\ 0 & -1/r^2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{5}$$

where r is the radius of the circle, and $\mathbf{f_3}$ is the transpose of the last row of F. A derivation of Equation (3) is given by Heikkilä [5].

Once the undistorted sensor coordinates of the ellipse centres have been computed, the image coordinates can be determined using the standard equations used in Tsai's model of the imaging process [8].

Scale cannot be inferred from the observations of the imaged control points alone: the positions of the cameras and the control points can be scaled by an arbitrary scale factor and remain consistent with the observations. In practice, the scale of the inferred parameters will be close to the true scale since the starting point is determined using an estimate of the 3D structure of the calibration object with approximately correct scale. In order to enforce correct scale, known distances must be used. The 2D structure of the patterns on the faces of the calibration objects are used to enforce scale. Knowledge of this 2D structure is limited to the accuracy of the laser printing process that is used to form the patterns. The 2D structure is used rather than the 3D structure, since it is likely to be known more accurately. Umeyamma's method [9] is used to determine the optimal rotation, translation and scaling in order to register the inferred 3D structure for each face with its expected structure (the 2D points are augmented with a $Z = 0$ coordinate since they are known to lie on a flat surface). This process of scaling, rotating and translating a data set of points to fit another data set of corresponding points is often referred to as Procrustes analysis[†]. The mean of the scaling factors corresponding to each face is used to scale the assumed radius of the circular targets, r, so that circle radii are in proportion to the size of the inferred 3D structure of the calibration object at each step of the bundle adjustment procedure. After the bundle adjustment is complete, the method is used to scale the camera positions and 3D calibration object structure appropriately. The internal camera parameters are unaffected by the scale: the focal lengths, principal points and radial distortion coefficients remain unchanged when a change in scale is imposed.

5 Results

The calibration procedure was tested using a cube with 50 mm sides and nine control points per face. The calibration object was machined from aluminium. A laser printer was used to print the coded target patterns onto adhesive paper. The patterns were then cut from the paper and stuck to the faces of the calibration object.

Since cubes can be manufactured highly accurately, the geometry of the inferred positions of the control points on the face of the cube can be used as a measure of the performance of the calibration procedure: inferred control point positions corresponding to the same face should be coplanar and sets of inferred positions corresponding to opposite faces should be an equal distance apart.

The camera system that was calibrated consisted of three approximately orthogonally mounted Dragonfly cameras. The Dragonfly cameras are grey-scale 640×480 pixel progressive scan cameras that are connected to an IEEE 1394 bus. The cameras are able to synchronise their acquisition time to within 20 μs.

The calibration procedure was coded in MATLAB. Where it was possible to achieve significant improvements in execution time, routines were coded as MEX (MATLAB executable) files in C. However, no particular effort was made to ensure fast execution and there is plenty of scope for reducing the running time of the calibration procedure. Since the calibration procedure is typically run off-line, and in principle need only be performed once, execution speed is of limited importance.

Twenty-four different poses of the cube were used. This resulted in 72 images from which 1423 imaged control points were located and identified. The approximate structure of the cube was determined using the known 2D structure of the six laser printed patterns for each cube face. Normally distributed noise with a standard deviation of 0.5 mm was added to each of the 3D coordinates to ensure that the calibration procedure produced correct results when the approximate 3D structure of the control points is known with only limited accuracy. The starting point parameters resulted in an RMS reprojection residual of 5.3 pixels. Bundle adjustment produced parameters which resulted in an RMS residual of 0.09 pixels.

The best-fit plane was determined for each of the six sets of nine control points corresponding to the six cube faces. The perpendicular distance between each inferred 3D control point position and its corresponding best fit-plane was then determined. The results are shown in Table 1. Note that since three points define a plane, one must multiply the values in Table 1 by $\frac{9}{9-3} = 1.5$ in order to obtain the expected RMS deviations that one would obtain as the

[†]Procrustes was a mythological Greek robber who offered travellers hospitality on a magical bed that would fit any guest. He would then stretch or shorten the travellers so that they fitted the bed perfectly.

Face	RMS Distance	Face	RMS Distance
1	2.9 μm	4	2.2 μm
2	1.6 μm	5	4.5 μm
3	1.9 μm	6	4.7 μm

Table 1: RMS distance to best-fit plane for sets of nine control points.

number of control points per face tends towards infinity. The observed deviations from the best-fit planes are due to errors in the calibration parameters and to the actual deviations from flatness of the control points on the cube (since neither the aluminium cube nor the adhesive paper is perfectly flat). The results shown in Table 1 thus represent an upper bound on the error that is associated with the 3D positions determined by the calibration procedure.

An indication of the accuracy of the structure can also be determined by computing the angles between the best fit planes. The angles were found to differ by a mean of 0.04° from what would be expected if the control points lay on the surface of a perfect cube.

Opposite Face Pairs	Estimate 1	Estimate 2
Face 1 and 6	50.1223 mm	50.1220 mm
Face 2 and 4	50.1317 mm	50.1317 mm
Face 3 and 5	50.1252 mm	50.1251 mm

Table 2: Estimates of the distance between opposite face pairs of the calibration cube.

The distances between opposite cube faces was calculated by determining the distance between the mean of a set of nine control points and the best-fit plane of the opposite face. This results in two estimates of the distances between each opposite face pair since the opposite best-fit planes are not perfectly parallel. Table 2 shows the results. The distances indicate that the thickness of the adhesive paper is between 60 μm and 70 μm, since the aluminium cube has 50 mm sides. The standard deviations of each of the three estimates gives a measure of the extent to which the three parallel face pairs are not the same distance apart. The standard deviations are 4.8 μm for the first set of estimates and 5.0 μm for the second.

The method for enforcing scale that is described in the previous section involves determining the best fit of the known 2D patterns to sets of inferred 3D control point positions that correspond to the same face. The differences between these two points sets, once the optimal similarity transform has been applied, is mainly due to inaccuracies in the laser printing process (calibration errors and the extent to which the surfaces are not perfectly flat are only minor contributors towards these differences). A measure of the accuracy of the laser printing process can be computed from these RMS differences. The RMS difference for all control points was measured to be 67 μm. This is the same order of magnitude as the dot size ($1\,\text{in}/300 \approx 85\,\mu$m) of a 300 dpi laser printer.

The execution time of the calibration procedure was roughly 2 hours on a 300 MHz Pentium II machine, using MATLAB 6.0 and the Windows NT operating system. Approximately 70% of this time was spent locating and identifying targets.

6 Conclusions

A complete calibration procedure for determining internal and pose parameters for a multiple camera system been described. Unlike other calibration procedures, the method is accurate, yet inexpensive since the 3D structure of the control points on the calibration object need not be accurately known in advance (obtaining a calibration object whose 3D structure is known in advance is usually a costly procedure). The control points also need not be visible from all viewpoints.

Coded target patterns are used so that control points can be located and identified automatically. Additional robustness is achieved by means of a RANSAC (*ran*dom *sa*mple *c*onsensus) algorithm that rejects any false positive target matches that do not conform to the consensus geometrical structure of the imaged control points (as defined by the approximately known 3D positions of the control points and the observed targets).

In order to reduce the effects of errors on the observed 2D positions of the imaged control points, and to ensure that the 3D structure of the calibration object can be determined along with the camera parameters, multiple poses of the calibration object are used.

A starting point parameter vector is determined by combining information derived from all of the images. This involves forming an initial estimate of the focal lengths of the cameras using Tsai's method, and then using a recently developed globally convergent pose estimation method to estimate camera and object poses. Multiple pose estimates are combined using an approach based on singular value decomposition. The starting point parameters are then refined using the Levenberg-Marquardt method to produce a set of parameters that is jointly optimal across all observations. To ensure accurate results, the imaging model takes into account the difference between an imaged circle-centre and an imaged-circle centre that occurs because of perspective distortion. Since scale cannot be inferred from image observations, it is derived from the structure of the flat patterns on each of the faces of the polyhedral calibration object.

Results indicate that reprojection residuals of less than one tenth of a pixel can be achieved across 72 images captured by three cameras. The inferred 3D structure of control points on the surface of an accurately machined cube conformed closely with the geometry of the points on the surface of a cube. Specifically, control points deviated by no more than 5 μm from the best fit planes corresponding to each of the faces. The angles between best fit planes and standard deviation of the distances between opposite face pairs were also very close to what would be obtained for control points lying on the faces of a perfect cube.

References

[1] S. J. Ahn, W. Rauh, and S. I. Kim. Circular coded target for automation of optical 3D-measurement and camera calibration. *International Journal of Pattern Recognition and Artificial Intelligence*, 15(6):905–919, 2001.

[2] T. A. Clarke and X. Wang. Extracting high precision information from CCD images. In *Optical methods and data processing for heat and fluid flow*, Proc. IMechE Conf., pages 311–320. City University, 1998.

[3] Thomas Coleman, Mary Ann Branch, and Andrew Grace. *Optimization Toolbox*. The Mathworks, January 1999.

[4] Andrew W. Fitzgibbon, Maurizio Pilu, and Robert B. Fisher. Direct least-squares fitting of ellipses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):476–480, May 1999.

[5] Janne Heikkilä. Geometric camera calibration using circular control points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10), October 2000.

[6] Chien-Ping Lu, Gregory D. Hager, and Eric Mjolsness. Fast and globally convergent pose estimation from video images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(6):610–622, June 2000.

[7] M. R. Shortis, T. A. Clarke, and T. Short. A comparison of some techniques for the subpixel location of discrete target images. In *Videometrics III*, volume 2350, pages 239–250. SPIE, 1994.

[8] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):232–344, August 1987.

[9] Shinji Umeyamma. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4), April 1991.

[10] R. J. Valkenburg. Camera calibration using multiple references. In *Image and Vision Computing New Zealand*, pages 61–66, August 1996.

[11] Zhengyou Zhang. A flexible new technique for camera calibration. Technical Report MSR-TR-98-71, Microsoft Research, December 1998.