

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Automated camera calibration for robot soccer

Donald G Bailey and Gourab Sen Gupta

*School of Engineering and Advanced Technology, Massey University
Palmerston North, New Zealand*

1. Introduction

Robot soccer has become popular over the last decade not only as a platform for education and entertainment but as a test bed for adaptive control of dynamic systems in a multi-agent collaborative environment (Messom, 1998). It is a powerful vehicle for exploration and dissemination of scientific knowledge in a fun and exciting manner. The robot soccer environment encompasses several technologies—embedded micro-controller based hardware, wireless radio-frequency data transmission, dynamics and kinematics of motion, motion control algorithms, real-time image capture and processing and multi-agent collaboration.

The vision system is an integral component of modern autonomous mobile robots. With robot soccer, the physical size of the robots in the micro-robot and small robot leagues limits the power and space available, precluding the use of local cameras on the robots themselves. This is overcome by using a global vision system, with one (or more for larger size fields) cameras mounted over the playing field. The camera or cameras are connected to a vision processor that determines the location and orientation of each robot and the location of the ball relative to the playing field. This data is then passed to the strategy controller, which determines how the team should respond to the current game situation, plans the trajectories or paths of the robots under its control, and transmits the appropriate low-level motor control commands to the robots, which enable them to execute the plan.

To manage complexity in collaborative robot systems, a hierarchical state transition based supervisory control (STBS) system can be used (Sen Gupta et al., 2002; Sen Gupta et al., 2004). However, the performance of such a system, or indeed any alternative higher-level control system, deteriorates substantially if the objects are not located accurately because the generic control functions to position and orient the robots are then no longer reliable.

The high speed and manoeuvrability of the robots make the game very dynamic. Accurate control of high-speed micro-robots is essential for success within robot soccer. This makes accurate, real-time detection of the position and orientation of objects of particular importance as these greatly affect path-planning, prediction of moving targets and obstacle avoidance. Each robot is identified in the global image by a “jacket” which consists of a pattern of coloured patches. The location of these coloured patches within the image is used to estimate the position and orientation of the robot within the playing area. The cameras must therefore be calibrated to provide an accurate mapping between image coordinates and world coordinates in terms of positions on the playing field (Bailey & Sen Gupta, 2004).

The portable nature of robot soccer platforms means that every time the system is set up, there are differences in the camera position and angle relative to the field. Each team has their own camera, and both cannot be mounted exactly over the centre of the playing area. It is also difficult to arrange the camera position so that it is perfectly perpendicular to the playing surface. Consequently, each camera is looking down on the playing area at a slight angle, which introduces a mild perspective distortion into the image. The size of the playing area, combined with constraints on how high the camera may be mounted, require that a wide angle lens be used. This can introduce significant barrel distortion within the image obtained. Both of these effects are readily apparent in Fig. 1. The limited height of the camera combined with the height of the robots means that each detected robot position is also subject to parallax error. These factors must all be considered, and compensated for, to obtain accurate estimates of the location and orientation of each robot.

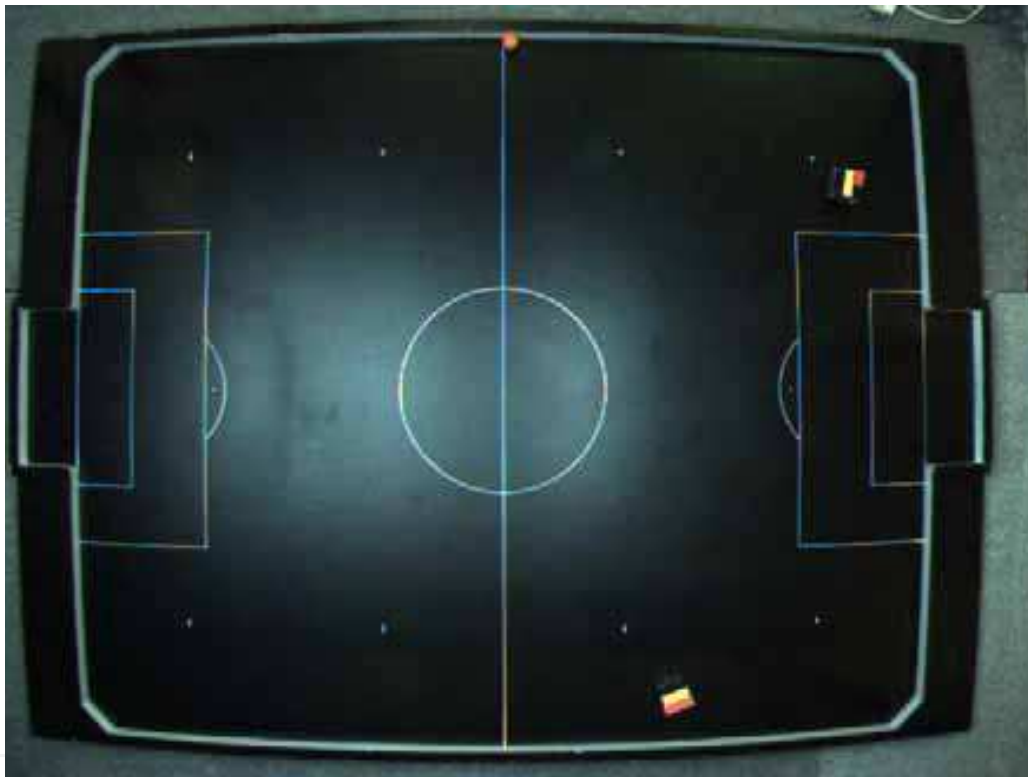


Fig. 1. Example field, clearly showing lens distortion and mild perspective distortion.

2. Effects of distortion

At the minimum, any calibration must determine the location of the playing field within the image. The simplest calibration (for single camera fields) is to assume that the camera is aligned with the field and that there is no distortion. The column positions of the goal mouths, C_{Left} and C_{Right} , and the row positions of the field edges at, or near, the centreline, R_{Top} and R_{Bottom} , need to be located within the image. Then, given the known length, L , and width, W , of the field, the estimated height of the camera, H , and the known height of the robots, h , an object positioned at row R and column C within the image may be determined relative to the centre of the field as

$$\begin{aligned}
 x &= \left(C - \frac{C_{Right} + C_{Left}}{2} \right) \left(\frac{L}{C_{Right} - C_{Left}} \right) \left(\frac{H-h}{H} \right) \\
 y &= \left(R - \frac{R_{Bottom} + R_{Top}}{2} \right) \left(\frac{W}{R_{Bottom} - R_{Top}} \right) \left(\frac{H-h}{H} \right)
 \end{aligned} \tag{1}$$

The first term sets the centre of the field as the origin, the second scales from pixel units to physical units, and the third term accounts for parallax distortion resulting from the height of the robot (assuming that the camera is positioned over the centre of the field). While this calibration is simple to perform, it will only be accurate for an ideal camera positioned precisely in the centre of the field and aligned perpendicularly to the playing field. Each deviation from ideal will introduce distortions in the image and calibration errors.

2.1 Lens distortion

The most prevalent form of lens distortion is barrel distortion. It results from the lens having a slightly higher magnification in the centre of the image than at the periphery. Barrel distortion is particularly noticeable with wide-angle lenses such as those used with robot soccer. While there are several different physical models of the lens distortion based on the known characteristics of the lens (Basu & Licardie, 1995; Pers & Kovacic, 2002), the most commonly used model is a generic radial Taylor series relating the ideal, undistorted coordinates (x_u, y_u) to the distorted coordinates in the image (x_d, y_d) :

$$\begin{aligned}
 x_d &= x_u \left(1 + \kappa_1 r_u^2 + \kappa_2 r_u^4 + \dots \right) + \left(\varepsilon_1 (r_u^2 + 2x_u^2) + 2\varepsilon_2 x_u y_u \right) \left(1 + \varepsilon_3 r_u^2 + \dots \right) \\
 y_d &= y_u \left(1 + \kappa_1 r_u^2 + \kappa_2 r_u^4 + \dots \right) + \left(2\varepsilon_1 x_u y_u + \varepsilon_2 (r_u^2 + 2y_u^2) \right) \left(1 + \varepsilon_3 r_u^2 + \dots \right)
 \end{aligned} \tag{2}$$

Both sets of coordinates have the centre of distortion as the origin, and $r_u^2 = x_u^2 + y_u^2$. The set of parameters κ and ε characterise a particular lens. Note that the centre of distortion is not necessarily the centre of the image (Willson & Shafer, 1994). For most lenses, two radial and two tangential terms are sufficient (Li & Lavest, 1996), and most calibration methods limit themselves to estimating these terms. A simple, one parameter, radial distortion model is usually sufficient to account for most of the distortion (Li & Lavest, 1996):

$$r_d = r_u (1 + \kappa r_u^2) \tag{3}$$

This forward transform is most commonly used because in modelling the imaging process, the image progresses from undistorted coordinates to distorted image coordinates. Sometimes, however, the reverse transform is used. This swaps the roles of the two sets of coordinates. Since the model is an arbitrary Taylor series expansion, either approach is valid (although the coefficients will be different for the forward and reverse transforms). The first order reverse transform is given by

$$r_u = r_d (1 + \kappa r_d^2) \tag{4}$$

Effect on position

If the simple calibration was based on distances from the centre of the image, the position error would increase with radius according to the radially dependent magnification. However, since the calibration of eq. (1) sets the positions at the edges and ends of the field, the position error there will be minimal. The absolute position errors should also be zero near the centre of the image, increase with radius to a local maximum between the centre and edges of the playing area, and decrease to zero again on an ellipse through the table edge and goal points. Outside this ellipse, the errors will increase rapidly with distance, having the greatest effect in the corners of the playing field.

Effect on orientation

Determining the effect of radial distortion on the angle is more complicated. Consider a point in the undistorted image using radial coordinates (r_u, φ) . At this point, the lens distortion results in a magnification

$$M = \frac{r_d}{r_u} \quad (5)$$

Next, consider a test point offset from this by a small distance, r , at an angle, θ_u , with a magnification, M_2 . If the magnification is constant ($M_2=M$) then everything is scaled equally, and by similar triangles, the angle to the test point will remain the same. This implies that if the test point is in the tangential direction ($\theta_u - \varphi = 90^\circ$), there will be no angle error. Similarly, since the distortion is radial, if the test point is aligned ($\theta_u - \varphi = 0$) the test point will be stretched radially, but the angle will not change. These two considerations imply that it is best to consider offsets in the tangential and radial direction. This geometry is shown in Fig. 2.

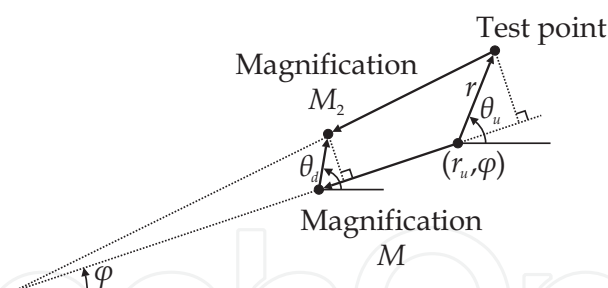


Fig. 2. Geometry for calculating the change in angle as a result of radial lens distortion. The distortion and scales have been exaggerated to make the effects visible.

After distortion, the angle to the test point becomes

$$\begin{aligned} \tan(\theta_d - \varphi) &= \frac{M_2 r \sin(\theta_u - \varphi)}{M_2 (r_u + r \cos(\theta_u - \varphi)) - M r_u} \\ &= \frac{M_2 r \sin(\theta_u - \varphi)}{M_2 r \cos(\theta_u - \varphi) + (M_2 - M) r_u} \end{aligned} \quad (6)$$

The test distance, r , is small, therefore

$$\begin{aligned}
 M_2 - M &\approx \Delta r_u^2 \frac{dM}{dr_u^2} \\
 &= \left((r_u - r \cos(\theta_u - \varphi))^2 - r_u^2 \right) \frac{dM}{dr_u^2} \\
 &\approx 2rr_u \cos(\theta_u - \varphi) \frac{dM}{dr_u^2}
 \end{aligned} \tag{7}$$

so

$$\tan(\theta_d - \varphi) \approx \frac{M}{M + 2r_u^2 \frac{dM}{dr_u^2}} \tan(\theta_u - \varphi) \tag{8}$$

When the forward map of eq. (3) is used, eq. (8) becomes

$$\tan(\theta_d - \varphi) \approx \frac{1 + \kappa r_u^2}{1 + 3\kappa r_u^2} \tan(\theta_u - \varphi) \tag{9}$$

Since the magnification changes faster with increasing radius, the angle error $(\theta_d - \theta_u)$ will also be larger further from the centre of distortion, and increase more rapidly in the periphery.

2.2 Perspective distortion

Perspective distortion results when the line of sight of the camera is not perpendicular to the plane of the playing area. This will occur when the camera is not directly over the centre of the playing area, and it must be tilted to fit the complete playing area within the field of view. A perspective transformation is given by

$$\begin{aligned}
 x_d &= \frac{h_1 x_u + h_2 y_u + h_3}{h_7 x_u + h_8 y_u + h_9} \\
 y_d &= \frac{h_4 x_u + h_5 y_u + h_6}{h_7 x_u + h_8 y_u + h_9}
 \end{aligned} \tag{10}$$

where (x_u, y_u) and (x_d, y_d) are the coordinates of an undistorted and distorted point respectively. This is often represented in matrix form using a homogenous coordinate system (Hartley & Zisserman, 2000):

$$\begin{bmatrix} kx_d \\ ky_d \\ k \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x_u \\ y_u \\ 1 \end{bmatrix} \text{ or } \mathbf{P}_d = \mathbf{H}\mathbf{P}_u \tag{11}$$

The 3x3 transformation matrix, \mathbf{H} , incorporates rotation, translation, scaling, skew, and stretch as well as perspective distortion. Just considering perspective distortion and for small tilt angles \mathbf{H} simplifies to

$$\mathbf{H}_{perspective} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ p_x & p_y & 1 \end{bmatrix} \quad (12)$$

The effect of this is to change the scale factor, k , in eq. (11), giving a position dependent magnification. As a consequence, parallel lines converge, meeting somewhere on the vanishing line given by

$$p_x x_u + p_y y_u + 1 = 0 \quad (13)$$

Effect on position

Since the simple calibration sets four points on the edges of the playing area, these points will have no error. In direction that the camera is tilted, k will be greater than 1, shrinking the scene. The radial error will be positive, and the tangential errors will be towards the direction line. In the opposite direction, the magnification is greater than 1. The radial error will be negative, and the tangential errors will be away from the direction line. There will be a line approximately across the middle of the playing area where

$$p_x x_u + p_y y_u + 1 = 1 \quad (14)$$

which will have no error. The angle of the line, and the severity of the errors will depend on the angle and extent of the camera tilt respectively.

Effect on orientation

Again the distortion will depend on the change of magnification with position. As the content becomes more compressed (closer to the vanishing line) angle distortion will increase, because the slope of the magnification becomes steeper and the angle errors will be greater. Without loss of generality, consider the camera tilted in the x direction ($p_y=0$). Consider a test point offset from (x_u, y_u) by distance r at an angle θ_u . After distortion, the offset becomes:

$$\begin{aligned} \Delta y &= \frac{y_u + r \sin \theta_u}{1 + p_x(x_u + r \cos \theta_u)} - \frac{y_u}{1 + p_x x_u} \\ \Delta x &= \frac{x_u + r \cos \theta_u}{1 + p_x(x_u + r \cos \theta_u)} - \frac{x_u}{1 + p_x x_u} \end{aligned} \quad (15)$$

Hence, the angle of the test point after distortion is given by

$$\begin{aligned} \tan \theta_d &= \frac{\Delta y}{\Delta x} = \frac{(y_u + r \sin \theta_u)(1 + p_x x_u) - y_u(1 + p_x(x_u + r \cos \theta_u))}{(x_u + r \cos \theta_u)(1 + p_x x_u) - x_u(1 + p_x(x_u + r \cos \theta_u))} \\ &= \frac{r \sin \theta_u (1 + p_x x_u) - y_u p_x r \cos \theta_u}{r \cos \theta_u} \\ &= (1 + p_x x_u) \tan \theta_u - y_u p_x \end{aligned} \quad (16)$$

Along the line of sight ($y_u=0$) there is no distortion radially or tangentially. However, other orientations become compressed as the magnification causes foreshortening, especially as it approaches the vanishing line. At other positions, angles that satisfy

$$\tan \theta_u = \frac{y_u}{x_u} \quad (17)$$

are not distorted. This is expected, since the perspective transformation will map straight lines onto straight lines. Orientations perpendicular to the line of sight ($\theta_u=90^\circ$) are not distorted. Other orientations are compressed by the perspective foreshortening. The mild perspective distortion encountered with the robot soccer system will introduce mild distortion. However, angle errors will be largest on the side of the field where the image appears compressed.

2.3 Parallax distortion

In the absence of any other information, the camera is assumed to be at a known height, H , directly above the centre of the robot soccer playing area. This allows the change in scale associated with the known heights of the robots to be taken into account. Since the robot jackets are always a fixed height above the playing surface, parallax correction simply involves scaling the detected position relative to the position of the camera. Errors in estimating the camera position will only introduce position errors; they will not affect the angle.

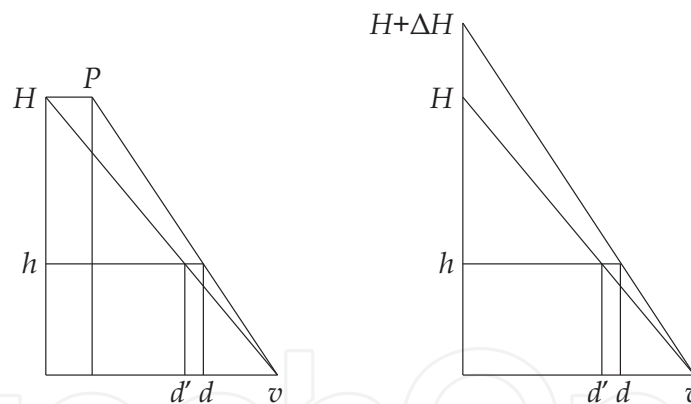


Fig. 3. Parallax correction geometry. Left: the effect of lateral error; right: the effect of height error. Note, the robot height, h , has been exaggerated for clarity.

Effect on position

Consider the geometry shown in Fig. 3. When the camera is offset laterally by P , an object at location d in the playing area will appear at location v by projecting the height

$$v = \frac{(d - P)H}{H - h} + P = \frac{dH - Ph}{H - h} \quad (18)$$

However, if it is assumed that the camera is not offset, the parallax correction will estimate the object position as d' . The error is given by

$$d' - d = \frac{v(H-h)}{H} - d = \frac{-Ph}{H} \quad (19)$$

The lateral error is scaled by the relative heights of the robot and camera. This ratio is typically 40 or 50, so a 5 cm camera offset will result in a 1 mm error in position. Note that the error applies to everywhere in the playing area, independent of the object location.

An error in estimating the height of the camera by ΔH will also result in an error in location of objects. In this case, the projection of the object position will be

$$v = \frac{d(H + \Delta H)}{H + \Delta H - h} \quad (20)$$

Again, given the assumptions in camera position, correcting this position for parallax will result in an error in estimating the robot position of

$$d' - d = \frac{v(H-h)}{H} - d = \frac{-dh\Delta H}{(H + \Delta H - h)H} \quad (21)$$

Since changing the height of the camera changes the parallax correction scale factor, the error will be proportional to the distance from the camera location. There will be no error directly below the camera, and the greatest errors will be seen in the corners of the playing area.

2.4 Effects on game play

When considering the effects of location and orientation errors on game play, two situations need to be considered. The first is local effects, for example when a robot is close to the ball and manoeuvring to shoot the ball. The second is when the robot is far from play, but must be brought quickly into play.

In the first situation, when the objects are relatively close to one another, what is most important is the relative location of the objects. Since both objects will be subject to similar distortions, they will have similar position errors. However, the difference in position errors will result in an error in estimating the angle between the objects (indeed this was how angle errors were estimated earlier in this section). While orientation errors may be considered of greater importance, these will correlate with the angle errors from estimating the relative position, making orientation errors less important for close work.

In contrast with this, at a distance the orientation errors are of greater importance, because shooting a ball or instructing the robot to move rapidly will result in moving in the wrong direction when the angle error is large. For slow play, this is less significant, because errors can be corrected over a series of successive images as the object is moving. However at high speed (speeds of over two metres per second are frequently encountered in robot soccer), estimating the angles at the start of a manoeuvre is more critical.

Consequently, good calibration is critical for successful game play.

3. Standard calibration techniques

In computer vision, the approach of Tsai (Tsai, 1987) or some derivation is commonly used to calibrate the relationship between pixels and real-world coordinates. These approaches

estimate the position and orientation of the camera relative to a target, as well as estimating the lens distortion parameters, and the intrinsic imaging parameters. Calibration requires a dense set of calibration data points scattered throughout the image. These are usually provided by a 'target' consisting of an array of spots, a grid, or a checkerboard pattern. From the construction of the target, the relative positions of the target points are well known. Within the captured image of the target, the known points are located and their correspondence with the object established. A model of the imaging process is then adjusted to make the target points match their measured image points.

The known location of the model enables target points to be measured in 3D world coordinates. This coordinate system is used as the frame of reference. A rigid body transformation (rotation and translation) is applied to the target points. This uses an estimate of the camera pose (position and orientation in world coordinates) to transform the points into a camera centred coordinate system. Then a projective transformation is performed, based on the estimated lens focal length, giving 2D coordinates on the image plane. Next, these are adjusted using the distortion model to account for distortions introduced by the lens. Finally, the sensing element size and aspect ratio are used to determine where the control points should appear in pixel coordinates. The coordinates obtained from the model are compared with the coordinates measured from the image, giving an error. The imaging parameters are then adjusted to minimise the error, resulting in a full characterisation of the imaging model.

The camera and lens model is sufficiently non-linear to preclude a simple, direct calculation of all of the parameters of the imaging model. Correcting imaging systems for distortion therefore requires an iterative approach, for example using the Levenberg-Marquardt method of minimising the mean squared error (Press et al., 1993). One complication of this approach is that for convergence, the initial estimates of the model parameters must be reasonably close to the final values. This is particularly so with the 3D rotation and perspective transformation parameters.

Planar objects are simpler to construct accurately than full 3D objects. Unfortunately, only knowing the location of points on a single plane is insufficient to determine a full imaging model (Sturm & Maybank, 1999). Therefore, if a planar target is used, several images must be taken of the target in a variety of poses to obtain full 3D information (Heikkila & Silven, 1996). Alternatively, a reduced model with one or two free parameters may be obtained from a single image. For robot soccer, this is generally not too much of a problem since the game is essentially planar.

A number of methods for performing the calibration for robot soccer are described in the literature. Without providing a custom target, there are only a few data points available from the robot soccer platform. The methods range from the minimum calibration described in the previous section through to characterisation of full models of the imaging system.

The basic approach described in section 2 does not account for any distortions. A simple approach was developed in (Weiss & Hildebrand, 2004) to account for the gross characteristics of the distortion. The playing area was divided into four quadrants, based on the centreline, and dividing the field in half longitudinally between the centres of the goals. Each quadrant was corrected using bilinear interpolation. While this corrects the worst of the position errors resulting from both lens and perspective distortion, it will only partially correct orientation errors. The use of a bilinear transformation will also result in a small jump in the orientation at the boundaries between adjacent quadrants.

A direct approach of Tsai's calibration is to have a chequered cloth (as the calibration pattern) that is rolled out over the playing area (Baltes, 2000). The corners of the squares on the cloth provide a 2D grid of target points for calibration. The cloth must cover as much as possible of the field of view of the camera. A limitation of this approach is that the calibration is with respect to the cloth, rather than the field. Unless the cloth is positioned carefully with respect to the field, this can introduce other errors.

This limitation may be overcome by directly using landmarks on the playing field as the target locations. This approach is probably the most commonly used and is exemplified in (Ball et al., 2004) where a sequence of predefined landmarks is manually clicked on within the image of the field. Tsai's calibration method is then used to determine the imaging model by matching the known locations with their image counterparts. Such approaches based on manually selecting the target points within the image are subject to the accuracy and judgement of the person locating the landmarks within the image. Target selection is usually limited to the nearest pixel. While selecting more points will generally result in a more accurate calibration by averaging the errors from the over-determined system, the error minimisation cannot remove systematic errors. Manual landmark selection is also very time-consuming.

The need to locate target points subjectively may be overcome by automating the calibration procedure. Egorova (Egorova et al., 2005) uses the bounding box to find the largest object in the image, and this is used to initialise the transform. A model of the field is transformed using iterative global optimisation to make the image of the field match the transformed model. While automatic, this procedure takes five to six seconds using a high end desktop computer for the model parameters to converge.

A slightly different approach is taken by Klancar (Klancar et al., 2004). The distortion correction is split into two stages: first the lens distortion is removed, and then the perspective distortion parameters are estimated. This approach to lens distortion correction is based on the observation that straight lines are invariant under a perspective (or projective) transformation. Therefore, any deviation from straightness must be due to lens distortion (Brown, 1971; Fryer et al., 1994; Park & Hong, 2001). This is the so-called 'plumb-line' approach, so named because when it was first used by (Brown, 1971), the straight lines were literally plumb-lines hung within the image. (Klancar et al., 2004) uses a Hough transform to find the major edges of the field. Three points are found along each line: one on the centre and one at each end. A hyperbolic sine radial distortion model is used (Pers & Kovacic, 2002), with the focal length optimised to make the three target points for each line as close to collinear as possible. One limitation of Klancar's approach is the assumption that the centre of the image corresponds with the centre of distortion. However, errors within the location of the distortion centre results in tangential distortion terms (Stein, 1997) which are not considered with the model. The second stage of Klancar's algorithm is to use the convergence of parallel lines (at the vanishing points) to estimate the perspective transformation component.

None of the approaches explicitly determines the camera location. Since they are all based on 2D targets, they can only gain limited information on the camera height, resulting in a limited ability to correct for parallax distortion. The limitations of the existing techniques led us to develop an automatic method that overcomes these problems by basing the calibration on a 3D model.

4. Automatic calibration procedure

The calibration procedure is based on the principles first described in (Bailey, 2002). A three stage solution is developed based on the 'plumb-line' principle. In the first stage, a parabola is fitted to each of the lines on the edge of the field. Without distortion, these should be straight lines, so the quadratic component provides data for estimating the lens distortion. A single parameter radial distortion model is used, with a closed form solution given for determining the lens distortion parameter. In the second stage, homogenous coordinates are used to model the perspective transformation. This is based on transforming the lines on the edge of the field to their known locations. The final stage uses the 3D information inherent in the field to obtain an estimate of the camera location (Bailey & Sen Gupta, 2008).

4.1 Edge detection

The first step is to find the edge of the playing field. The approach taken will depend on the form of the field. Our initial work was based on micro-robots, where the playing field is bounded by a short wall. The white edges apparent in Fig. 1 actually represent the inside edge of the wall around the playing area, as shown in Fig. 4. In this case, the edge of the playing area corresponds to the edge between the white of the wall and the black of the playing surface. While detecting the edge between the black and white sounds straightforward, it is not always as simple as that. Specular reflections off the black regions can severely reduce the contrast in some situations, as can be seen in Fig. 5, particularly in the bottom right corner of the image.

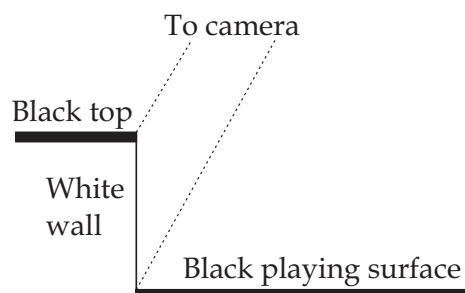


Fig. 4. The edge of the playing area.

Two 3x3 directional Prewitt edge detection filters are used to detect both the top and bottom edges of the walls on all four sides of the playing area. To obtain an accurate estimate of the calibration parameters, it is necessary to detect the edges to sub-pixel accuracy. Consider first the bottom edge of the wall along the side of the playing area in the top edge of the image. Let the response of the filtered image be $f[x,y]$. Within the top 15% of the image, the maximum filtered response is found in each column. Let the maximum in column x be located on row $y_{max,x}$. A parabola is fitted to the filter responses above and below this maximum (perpendicular to the edge), and the edge pixel determined to sub-pixel location as (Bailey, 2003):

$$edge[x] = y_{max,x} + \frac{f[x, y_{max,x} + 1] - f[x, y_{max,x} - 1]}{4f[x, y_{max,x}] - 2(f[x, y_{max,x} + 1] + f[x, y_{max,x} - 1])} \quad (22)$$

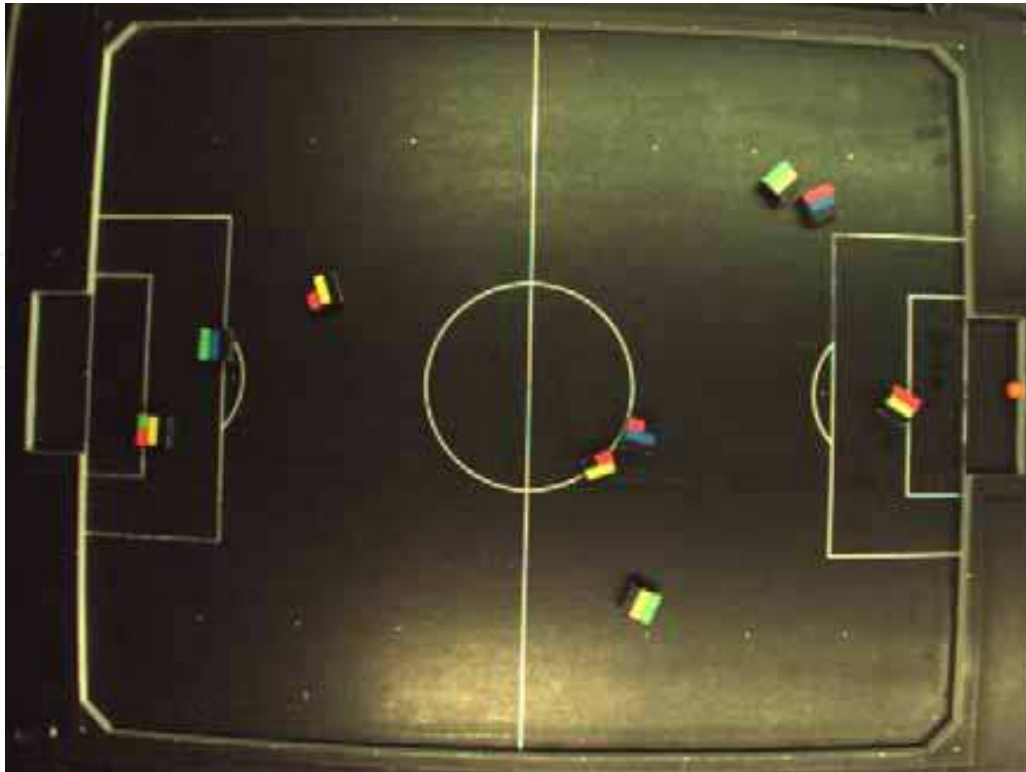


Fig. 5. As a result of lighting and specular reflection, the edge of the playing area may be harder to detect.

A parabola is then fitted to all the detected edge points $(x, edge[x])$ along the length of the edge. Let the parabola be $y(x) = ax^2 + bx + c$. The parabola coefficients are determined by minimising the squared error

$$E = \sum_x (ax^2 + bx + c - edge[x])^2 \quad (23)$$

The error is minimised by taking partial derivatives of eq. (23) with respect to each of the parameters a , b , and c , and solving for when these are equal to zero. This results in the following set of simultaneous equations, which are then solved for the parabola coefficients.

$$\begin{bmatrix} \sum x^4 & \sum x^3 & \sum x^2 \\ \sum x^3 & \sum x^2 & \sum x \\ \sum x^2 & \sum x & \sum 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \sum x^2 edge[x] \\ \sum x edge[x] \\ \sum edge[x] \end{bmatrix} \quad (24)$$

The resulting parabola may be subject to errors from noisy or misdetections. The accuracy may be improved considerably using robust fitting techniques. After initially estimating the parabola, any outliers are removed from the data set, and the parabola refitted to the remaining points. Two iterations are used, removing points more than 1 pixel from the parabola in the first iteration, and removing those more than 0.5 pixel from the parabola in the second iteration.

A similar process is used with the local minimum of the Prewitt filter to detect the top edge of the wall. The process is repeated for the other walls in the bottom, left and right edges of

the image. The robust fitting procedure automatically removes the pixels in the goal mouth from the fit. The results of detecting the edges for the image in Fig. 1 are shown in Fig. 6.

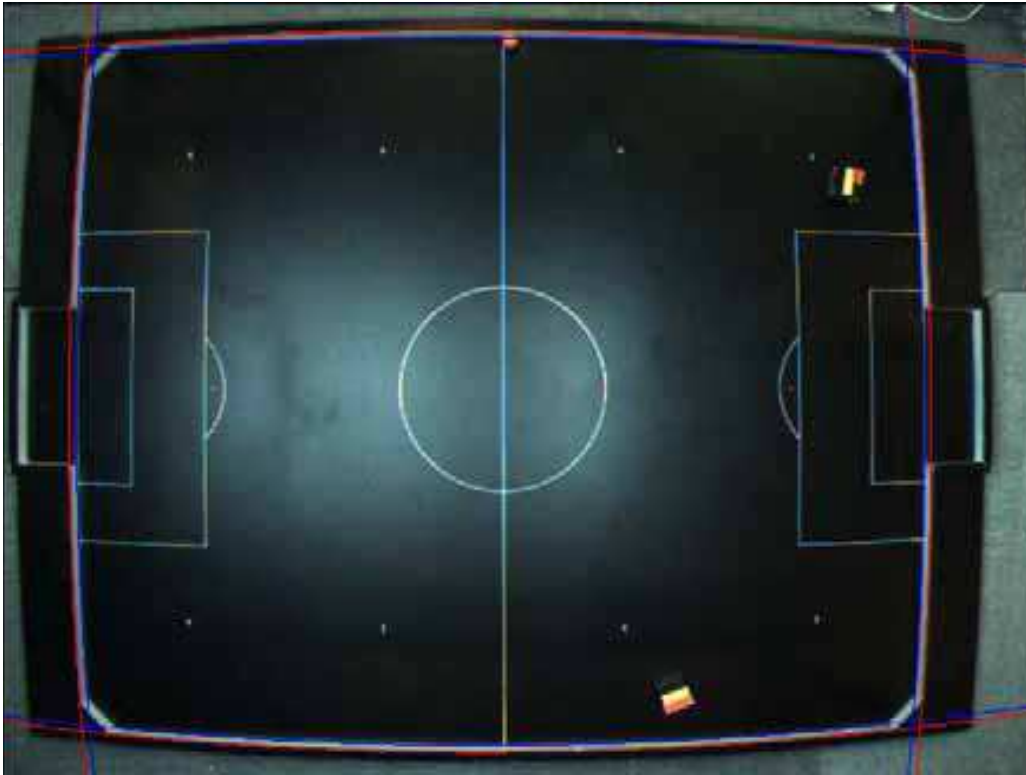


Fig. 6. The detected walls from the image in Fig. 1.

4.2 Estimating the distortion centre

Before correcting for the lens distortion, it is necessary to estimate the centre of distortion. With purely radial distortion, lines through the centre will remain straight. Therefore, considering the parabola components, a line through the centre of distortion will have no curvature ($a=0$). In general, the curvature of a line will increase the further it is from the centre. It has been found that the curvature, a , is approximately proportional to the axis intercept, c , when the origin is at the centre of curvature (Bailey, 2002).

The x centre, x_0 , maybe determined by considering the vertical lines within the image (the left and right ends of the field) and the y centre, y_0 , from the horizontal lines (the top and bottom sides of the field). Consider the horizontal centre first. With just two lines, one at each end of the field, the centre of distortion is given by

$$x_0 = \frac{a_2 c_1 - a_1 c_2}{a_2 - a_1} \quad (25)$$

With more than two lines available, this may be generalised by performing a least squares fit between the intercept and the curvature:

$$x_0 = \frac{\sum c_i a_i \sum c_i - \sum a_i \sum c_i^2}{\sum c_i a_i \sum 1 - \sum a_i \sum c_i} \quad (26)$$

The same equations may be used to estimate the y position of the centre, y_0 . Once the centre has been estimated, it is necessary to offset the parabolas to make this the origin. This involves substituting

$$\begin{aligned}\hat{x} &= x - x_0 \\ \hat{y} &= y - y_0\end{aligned}\quad (27)$$

into the equations for each parabola, $y = ax^2 + bx + c$ to give

$$\begin{aligned}\hat{y} &= a(\hat{x} + x_0)^2 + b(\hat{x} + x_0) + c - y_0 \\ &= a\hat{x}^2 + (2ax_0 + b)\hat{x} + (ax_0^2 + bx_0 + c - y_0)\end{aligned}\quad (28)$$

and similarly for $x = ay^2 + by + c$ with the x and y reversed.

Shifting the origin changes the parabola coefficients. In particular, the intercept changes, as a result of the curvature and slope of the parabolas. Therefore, this step is usually repeated two or three times to progressively refine the centre of distortion. The centre relative to the original image is then given by the sum of successive offsets.

4.3 Estimating the aspect ratio

For pure radial distortion, the slopes of the a vs c curve should be the same horizontally and vertically. This is because the strength of the distortion depends only on the radius, and not on the particular direction. When using an analogue camera and frame grabber, the pixel clock of the frame grabber is not synchronised with the pixel clock of the sensor. Any difference in these clock frequencies will result in aspect ratio distortion with the image stretched or compressed horizontally by the ratio of the clock frequencies. This distortion is not usually a problem with digital cameras, where the output pixels directly correspond to sensing elements. However, aspect ratio distortion can also occur if the pixel pitch is different horizontally and vertically.

To correct for aspect ratio distortion if necessary, the x axis can be scaled as $\hat{x} = x/R$. The horizontal and vertical parabolas are affected by this transformation in different ways:

$$\begin{aligned}y &= ax^2 + bx + c \\ &= aR^2\hat{x}^2 + bR\hat{x} + c\end{aligned}\quad (29)$$

and

$$\hat{x} = \frac{x}{R} = \frac{a}{R}y^2 + \frac{b}{R}y + \frac{c}{R}\quad (30)$$

respectively. The scale factor, R , is chosen to make the slopes of a vs c to be the same horizontally and vertically. Let s_x be the slope of a vs c for the horizontal parabolas and s_y be the slope for the vertical parabolas. The scale factor is then given by

$$R = \sqrt{s_x/s_y}\quad (31)$$

4.4 Estimating the lens distortion parameter

Since the aim is to transform from distorted image coordinates to undistorted coordinates, the reverse transform of eq. (4) is used in this work. Consider first a distorted horizontal line. It is represented by the parabola $y_d = ax_d^2 + bx_d + c$. The goal is to select the distortion parameter, κ , that converts this to a straight line. Substituting this into eq. (4) gives

$$\begin{aligned} y_u &= y_d(1 + \kappa(x_d^2 + y_d^2)) \\ &= (ax_d^2 + bx_d + c)\left(1 + \kappa\left(x_d^2 + (ax_d^2 + bx_d + c)^2\right)\right) \\ &= c(1 + \kappa c^2) + b(1 + 3\kappa c^2)x_d + (a + c\kappa(3ac + 3b^2 + 1))x_d^2 + \dots \end{aligned} \quad (32)$$

where the ... represents higher order terms. Unfortunately, this is in terms of x_d rather than x_u . If we consider points near the centre of the image (small x) then the higher order terms are negligible so

$$\begin{aligned} x_u &= x_d(1 + \kappa x_d^2) \\ &\approx x_d(1 + \kappa y_d^2) \\ &\approx x_d(1 + \kappa c^2) \end{aligned} \quad (33)$$

or

$$x_d \approx \frac{x_u}{1 + \kappa c^2} \quad (34)$$

Substituting this into eq. (32) gives

$$y_u = c(1 + \kappa c^2) + \frac{b(1 + 3\kappa c^2)}{1 + \kappa c^2}x_u + \frac{a + c\kappa(3ac + 3b^2 + 1)}{(1 + \kappa c^2)^2}x_u^2 + \dots \quad (35)$$

Again, assuming points near the centre of the image, and neglecting the higher order terms, eq. (35) will be a straight line if the coefficient of the quadratic term is set to zero. Solving this for κ gives

$$\kappa = \frac{-a}{c(3ac + 3b^2 + 1)} \quad (36)$$

Each parabola (in both horizontal and vertical directions) will give separate estimates of κ . These are simply averaged to get a value of κ that works reasonably well for all lines. (Note that if there are any lines that pass close to the origin, a weighted average should be used because the estimate of κ from such lines is subject to numerical error (Bailey, 2002).)

Setting the quadratic term to zero, and ignoring the higher order terms, each parabola becomes a line

$$\begin{aligned} y_u &= \frac{b(1 + 3\kappa c^2)}{1 + \kappa c^2}x_u + c(1 + \kappa c^2) \\ &= m_y x_u + d_y \end{aligned} \quad (37)$$

and similarly for the vertical lines. The change in slope of the line at the intercept reflects the angle distortion and is of a similar form to eq. (9). Although the result of eq. (37) is based on the assumption of points close to the origin, in practise, the results are valid even for quite severe distortions (Bailey, 2002).

4.5 Estimating the perspective transformation

After correcting for lens distortion, the edges of the playing area are straight. However, as a result of perspective distortion, opposite edges may not necessarily be parallel. The origin is also at the centre of distortion, rather than in more convenient field-centric coordinates. This change of coordinates may involve translation and rotation in addition to just a perspective map. Therefore the full homogenous transformation of eq. (11) will be used. The forward transformation matrix, \mathbf{H} , will transform from undistorted to distorted coordinates. To correct the distortion, the reverse transformation is required:

$$\mathbf{P}_u = \mathbf{H}^{-1}\mathbf{P}_d \quad (38)$$

The transformation matrix, \mathbf{H} , and its inverse \mathbf{H}^{-1} , have only 8 degrees of freedom since scaling \mathbf{H} by a constant will only change the scale factor k , but will leave the transformed point unchanged. Each line has two parameters, so will therefore provide two constraints on \mathbf{H} . Therefore, four lines, one from each side of the playing field, are sufficient to determine the perspective transformation.

The transformation of eq. (38) will transform points rather than lines. The line (from eq. (37)) may be represented using homogenous coordinates as

$$\begin{bmatrix} m_y & -1 & d_y \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0 \text{ or } \mathbf{LP} = 0 \quad (39)$$

where \mathbf{P} is a point on the line. The perspective transform maps lines onto lines, therefore a point on the distorted line ($\mathbf{L}_d\mathbf{P}_d=0$) will lie on the transformed line ($\mathbf{L}_u\mathbf{P}_u=0$) after correction. Substituting into eq. (11) gives

$$\mathbf{L}_u = \mathbf{L}_d\mathbf{H} \quad (40)$$

The horizontal lines, $y = m_y x + d_y$, need to be mapped to their known location on the sides of the playing area, at $y=Y$. Substituting into eq. (40) gives three equations in the coefficients of \mathbf{H} :

$$\begin{aligned} 0 &= m_y h_1 - h_2 + d_y h_3 \\ -1 &= m_y h_4 - h_5 + d_y h_6 \\ Y &= m_y h_7 - h_8 + d_y h_9 \end{aligned} \quad (41)$$

Although there are 3 equations, there are only two independent equations. The first equation constrains the transformed line to be horizontal. The last two, taken together, specify the vertical position of the line. The two constraint equations are therefore

$$\begin{aligned} 0 &= m_y h_1 - h_2 + d_y h_3 \\ 0 &= Y m_y h_4 - Y h_5 + Y d_y h_6 + m_y h_7 - h_8 + d_y h_9 \end{aligned} \quad (42)$$

Similarly, the vertical lines, $x = m_x y + d_x$, need to be mapped to their known locations at the ends of the field, at $x=X$.

$$\begin{aligned} 0 &= -h_4 + m_x h_5 + d_x h_6 \\ 0 &= -X h_1 + X m_x h_2 + X d_x h_3 - h_7 + m_x h_8 + d_x h_9 \end{aligned} \quad (43)$$

For the robot soccer platform, each wall has two edges. The bottom edge of the wall maps to the known position on the field. The bottom edge of each wall will therefore contribute two equations. The top edge of the wall, however, is subject to parallax, so its absolute position in the 2D reference is currently unknown. However, it should still be horizontal or vertical, as represented by the first constraint of eq. (42) or (43) respectively. These 12 constraints on the coefficients of \mathbf{H} can be arranged in matrix form (showing only one set of equations for each horizontal and vertical edge):

$$\begin{bmatrix} m_y & -1 & d_y & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_y Y & -Y & d_y Y & m_y & -1 & d_y \\ 0 & 0 & 0 & -1 & m_x & d_x & 0 & 0 & 0 \\ -X & m_x X & d_x X & 0 & 0 & 0 & -1 & m_x & d_x \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_9 \end{bmatrix} = \mathbf{0} \text{ or } \mathbf{D}\hat{\mathbf{H}} = \mathbf{0} \quad (44)$$

Finding a nontrivial solution to this requires determining the null-space of the 12x9 matrix, \mathbf{D} . This can be found through singular value decomposition, and selecting the vector corresponding to the smallest singular value (Press et al., 1993). The alternative is to solve directly using least squares. First, the square error is defined as

$$E = \mathbf{D}\hat{\mathbf{H}}(\mathbf{D}\hat{\mathbf{H}})^T = \mathbf{D}\hat{\mathbf{H}}\hat{\mathbf{H}}^T\mathbf{D}^T \quad (45)$$

Then the partial derivative is taken with respect to the coefficients of $\hat{\mathbf{H}}$:

$$\frac{\partial E}{\partial \hat{\mathbf{H}}} = \mathbf{D}^T \mathbf{D} \hat{\mathbf{H}} = \mathbf{0} \quad (46)$$

$\mathbf{D}^T \mathbf{D}$ is now a square 9x9 matrix, and $\hat{\mathbf{H}}$ has eight independent unknowns. The simplest solution is to fix one of the coefficients, and solve for the rest. Since the camera is approximately perpendicular to the playing area, h_9 can safely be set to 1. The redundant bottom line of $\mathbf{D}^T \mathbf{D}$ can be dropped, and the right hand column of $\mathbf{D}^T \mathbf{D}$ gets transferred to the right hand side. The remaining 8x8 system may be solved for h_1 to h_8 . Once solved, the elements are rearranged back into a 3x3 matrix for \mathbf{H} , and each of the lines is transformed to give two sets of parallel lines for the horizontal and vertical edges.

The result of applying the distortion correction to the input image is shown in Fig. 7.

4.6 Estimating the camera position

The remaining step is to determine the camera position relative to the field. While in principle, this can be obtained from the perspective transform matrix if the focal length and sensor size are known, here they will be estimated directly from measurements on the field. The basic principle is to back project the apparent positions of the top edges of the walls on two sides. These will intersect at the camera location, giving both the height and lateral position, as shown in Fig. 8.

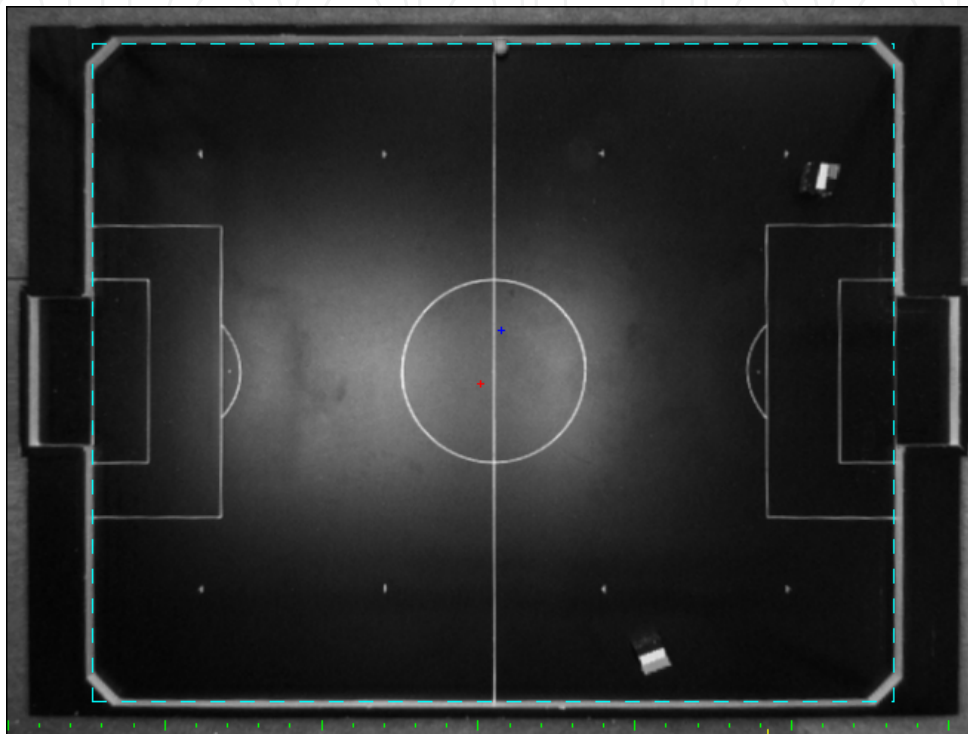


Fig. 7. The image after correcting for distortion. The blue + corresponds to the centre of distortion, and the red + corresponds to the detected camera position. The camera height is indicated in the scale on the bottom (10 cm per division).

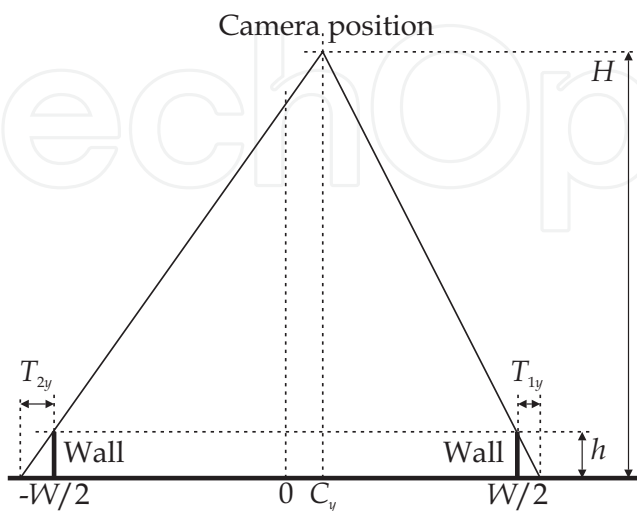


Fig. 8. Geometry for estimating the camera position.

The image from the camera can be considered as a projection of every object onto the playing field. Having corrected for distortion, the bottom edges of the walls will appear in their true locations, and the top edges of the walls are offset by parallax.

Let the width of the playing area be W and wall height be h . Also let the width of the projected side wall faces be T_{1y} and T_{2y} . The height, H , and lateral offset of the camera from the centre of the field, C_y , may be determined from similar triangles:

$$\frac{H}{\frac{W}{2} - C_y + T_{1y}} = \frac{h}{T_{1y}} \quad (47)$$

Rearranging gives:

$$T_{1y} = \frac{h\left(\frac{W}{2} - C_y\right)}{H - h} \quad (48)$$

and similarly for the other wall

$$T_{2y} = \frac{h\left(\frac{W}{2} + C_y\right)}{H - h} \quad (49)$$

Equations (48) and (49) can be solved to give the camera location

$$C_y = \left(\frac{T_{2y} - T_{1y}}{T_{2y} + T_{1y}}\right) \frac{W}{2} \quad (50)$$

$$H = \frac{hW}{T_{1y} + T_{2y}} + h \quad (51)$$

Similar geometrical considerations may be applied along the length of the field to give

$$C_x = \left(\frac{T_{2x} - T_{1x}}{T_{2x} + T_{1x}}\right) \frac{L}{2} \quad (52)$$

$$H = \frac{hL}{T_{1x} + T_{2x}} + h \quad (53)$$

where L is the length of the playing field and T_{1x} and T_{2x} are the width of the projected end walls.

Equations (50) to (53) give four independent equations for three unknowns. Measurement limitations and noise usually result in equations (51) and (53) giving different estimates of the camera height. In such situations, it is usual to determine the output values (C_x , C_y , and H) that are most consistent with the input data (T_{1x} , T_{2x} , T_{1y} , and T_{2y}). For a given camera location, the error between the corresponding input and measurement can be obtained from eq. (48) as

$$E_{1y} = \frac{h\left(\frac{W}{2} - C_y\right)}{H - h} - T_{1y} \quad (54)$$

and similarly for each of the other inputs. The camera location can then be chosen that minimises the total squared error

$$E^2 = E_{1y}^2 + E_{2y}^2 + E_{1x}^2 + E_{2x}^2 \quad (55)$$

This can be found by taking partial derivatives of eq. (55) with respect to each of the camera location variables and solving for the result to 0:

$$\frac{\partial E^2}{\partial C_y} = \frac{4h^2 C_y}{(H-h)^2} - \frac{2h(T_{2y} - T_{1y})}{H-h} = 0 \quad (56)$$

or

$$\frac{2hC_y}{H-h} = T_{2y} - T_{1y} \quad (57)$$

Similarly

$$\frac{\partial E^2}{\partial C_x} \Rightarrow \frac{2hC_x}{H-h} = T_{2x} - T_{1x} \quad (58)$$

The partial derivative with respect to the camera height is a little more complex because H appears in the denominator of each of the terms. The partial derivative of the errors across the width of the field is

$$\frac{\partial E_y^2}{\partial H} = \frac{\partial E_{1y}^2}{\partial H} + \frac{\partial E_{2y}^2}{\partial H} = \frac{-h}{(H-h)^2} \left(\frac{4h \left(\frac{W^2}{4} + C_y^2 \right)}{H-h} - W(T_{1y} + T_{2y}) - 2C_y(T_{2y} - T_{1y}) \right) \quad (59)$$

This can be simplified by eliminating C_y through substituting eq. (57)

$$\frac{\partial E_y^2}{\partial H} = \frac{-h}{(H-h)^2} \left(\frac{hW^2}{H-h} - W(T_{1y} + T_{2y}) \right) \quad (60)$$

Finally, combining the partial derivatives along the length of the field with those across the width of the field gives:

$$\frac{\partial E^2}{\partial H} = \frac{\partial E_y^2}{\partial H} + \frac{\partial E_x^2}{\partial H} = \frac{-h}{(H-h)^2} \left(\frac{hW^2}{H-h} - W(T_{1y} + T_{2y}) + \frac{hL^2}{H-h} - L(T_{1x} + T_{2x}) \right) = 0 \quad (61)$$

Solving for H gives

$$H = \frac{(W^2 + L^2)h}{W(T_{1y} + T_{2y}) + L(T_{1x} + T_{2x})} + h \quad (62)$$

Finally, the result from eq. (62) can be substituted into equations (57) and (58) to give the lateral position of the camera:

$$(C_x, C_y) = \frac{\frac{1}{2}(W^2 + L^2)}{W(T_{1y} + T_{2y}) + L(T_{1x} + T_{2x})} (T_{2x} - T_{1x}, T_{2y} - T_{1y}) \quad (63)$$

The detected position of the camera is overlaid on the undistorted image in Fig. 7.

5. Applying the corrections

While it is possible to apply the distortion correction to the image prior to detecting the objects, in practise this is computationally inefficient. Only a relatively small number of objects need to be detected, and the distortion is not so severe as to preclude reliable detection directly within the distorted image. Therefore, the robots and ball positions are detected within the distorted image, with the position (and orientation in the case of the robot players) returned in distorted image coordinates. The procedure for correcting the image coordinates follows the calibration procedure described in the previous section.

5.1 Correcting object position

First, the detected feature location (x_f, y_f) is offset relative to the centre of distortion from eq. (27) and corrected for aspect ratio distortion if necessary:

$$(x_d, y_d) = ((x_f - x_0)/R, y_f - y_0) \quad (64)$$

The lens distortion is then corrected by applying the radially dependent magnification from eq. (4)

$$(x_u, y_u) = (1 + \kappa r_d^2)(x_d, y_d) \quad (65)$$

This point is then transformed into field-centric coordinates and corrected for perspective distortion by applying eq. (38)

$$\begin{bmatrix} kx_p \\ ky_p \\ k \end{bmatrix} = \mathbf{H}^{-1} \begin{bmatrix} x_u \\ y_u \\ 1 \end{bmatrix} \quad (66)$$

where \mathbf{H}^{-1} is the inverse of the matrix obtained from solving eq. (46) in fitting the field edges. The resulting point is normalised by dividing through the left hand side of eq. (66) by k . Finally, the feature point is corrected for parallax error. From similar triangles

$$\frac{\hat{x}_f - C_x}{H - h_f} = \frac{x_p - C_x}{H} \quad (67)$$

where h_f is the known height of the object and (\hat{x}_f, \hat{y}_f) is the corrected location of the object feature point. Equation (67), and its equivalent in the y direction, may be rearranged to give the corrected feature location:

$$(\hat{x}_f, \hat{y}_f) = \frac{H - h_f}{H} (x_p, y_p) + \frac{h_f}{H} (C_x, C_y) \quad (68)$$

The first term in eq. (68) is the scale factor that corrects for the height of the object, and the second term compensates for the lateral position of the camera as in eq. (19).

5.2 Correcting object orientation

As outlined in section 2, the distortion will affect the detected orientation of objects within the image. The simplest approach to correct the object orientation, θ , is to also transform a test point (x_t, y_t) that is offset a small distance, r , from the object location in the direction specified by the orientation:

$$(x_t, y_t) = (x_f, y_f) + r(\cos \theta, \sin \theta) \quad (69)$$

The offset should be of similar order to the offset used to measure the orientation in the distorted image (for example half the width of the robot). The corrected orientation may then be determined from the angle between the corrected test point and the corrected object location:

$$\hat{\theta} = \tan^{-1} \left(\frac{\hat{y}_t - \hat{y}_f}{\hat{x}_t - \hat{x}_f} \right) \quad (70)$$

6. Results and discussion

As the image in Fig. 7 shows, the calibration method is effective at correcting distortion around the edge of the field. However, to have confidence that the model is actually correcting points anywhere in the playing area, it is necessary to check the transformation at a number of points scattered throughout the image. The calibration procedure was tested on three fields. The first was a small (150 cm x 130 cm) 3-aside micro-robot playing field, captured using a 320x240 analogue camera (Bailey & Sen Gupta, 2004). The second two were larger (220 cm x 180 cm) 5-aside fields, captured using a 656x492 digital Firewire camera (Bailey & Sen Gupta, 2008).

On the small field, a set of 76 points was extracted from throughout the playing area using the field lines and free kick markers as input, as indicated in Fig. 9. The RMS residual error after correcting the validation points was 1.75 mm, which corresponds to about 30% of the width of a pixel. The lateral position of the camera was in error by 1.2 cm, which results in a negligible parallax error (from eq. (19)). The height of the camera was over-estimated by approximately 9 cm. This will give a maximum parallax error in the corners of the playing field (from eq. (21)) of approximately 1.1 mm, which is again a fraction of a pixel.

For the larger fields (shown in Fig. 1 and Fig. 5) both resulted in a significantly improved image that appeared to be free from major distortions (see Fig. 7). In both cases, the lateral position of the camera was also measured with good accuracy, with a total lateral error of 0.9 cm and 1.0 cm respectively for the two fields. Again, the consequent parallax error (from eq. (19)) is negligible. The error in the height, however, was significantly larger, with an under-estimate of 8.6 cm for the image in Fig. 1 and an under-estimate of 33 cm for the image in Fig. 5. Even this large height error results in an error of less than 3 mm in the corners of the field (from eq. (21)). This is still less than one pixel at the resolution of the image, and the error is significantly smaller over the rest of the field.

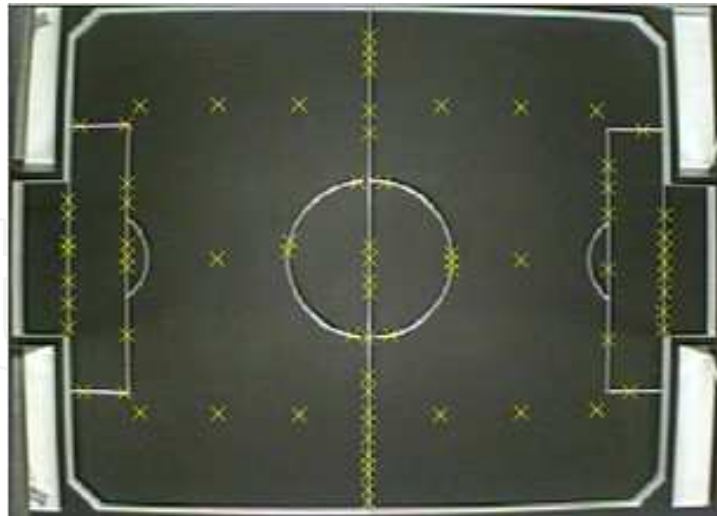


Fig. 9. The 3-side field with validation points marked.

The larger height errors are not completely unexpected, because the height is estimated by back projecting the relatively small parallax resulting from a low wall. The wall only occupied 3 pixels in the image of the small field and approximately 5 pixels in the larger field. Measurement of this parallax requires sub-pixel accuracy to gain any meaningful results. Any small errors in measuring the wall parallax are amplified to give a large error in the estimate of the camera height. The lateral position is not affected by measurement errors to the same extent, because it is based on the relative difference in parallax between the two sides.

The cause of the large height error was examined in some detail in (Bailey & Sen Gupta, 2008). The under-estimate of the height was caused by the measured parallax of the walls being larger than expected (although the error was still sub-pixel). Since the parallax appears in the denominator of eq. (62), any over-estimate of the parallax will result in an under-estimate of the height of the camera. Two factors contributed to this error. First, a slight rounding of the profile at the top of the wall combined with specular reflection resulted in the boundary between the white and black extending over the top of the wall, increasing the apparent width. A second factor, which exacerbates this in Fig. 5, is that the position of the lights gave a stronger specular component in the vicinity of the walls.

The other fields were less affected for the following reasons. Firstly, the lights were positioned over the field rather than outside it. The different light angle means that these fields were less prone to the specular reflection effects on the top corner of the wall. Secondly, the other fields were in a better condition, having been repainted more recently, and with less rounding of the top edge of the walls. Consequently, the wall parallax was able to be measured more accurately and the resultant errors were less significant.

6.1 Future work

The next step is to extend the work presented to the larger 11-sided field. These fields have two cameras, one over each half of the field. The calibration principles will be the same. The biggest difference is that the centreline will form one of the edges of the calibration, and this does not have a height associated with it. This will limit the accuracy of the parallax

correction data, although in principle the height of three walls should provide sufficient data for estimating the camera location.

A further extension is to the Robocup league, which has no walls. Again two cameras are required, one to capture each half of the field, as shown in Fig. 10. The image processing algorithms will need to be modified for line detection rather than edge detection, and another mechanism found to estimate the camera position. One approach currently being experimented with is to place poles of known height in each corner and at each end of the centre-line as can be seen in Fig. 10. The parabola based lens and perspective distortion correction will be based on the edges of the field and centreline, and the markers on the poles detected and back projected to locate the camera.

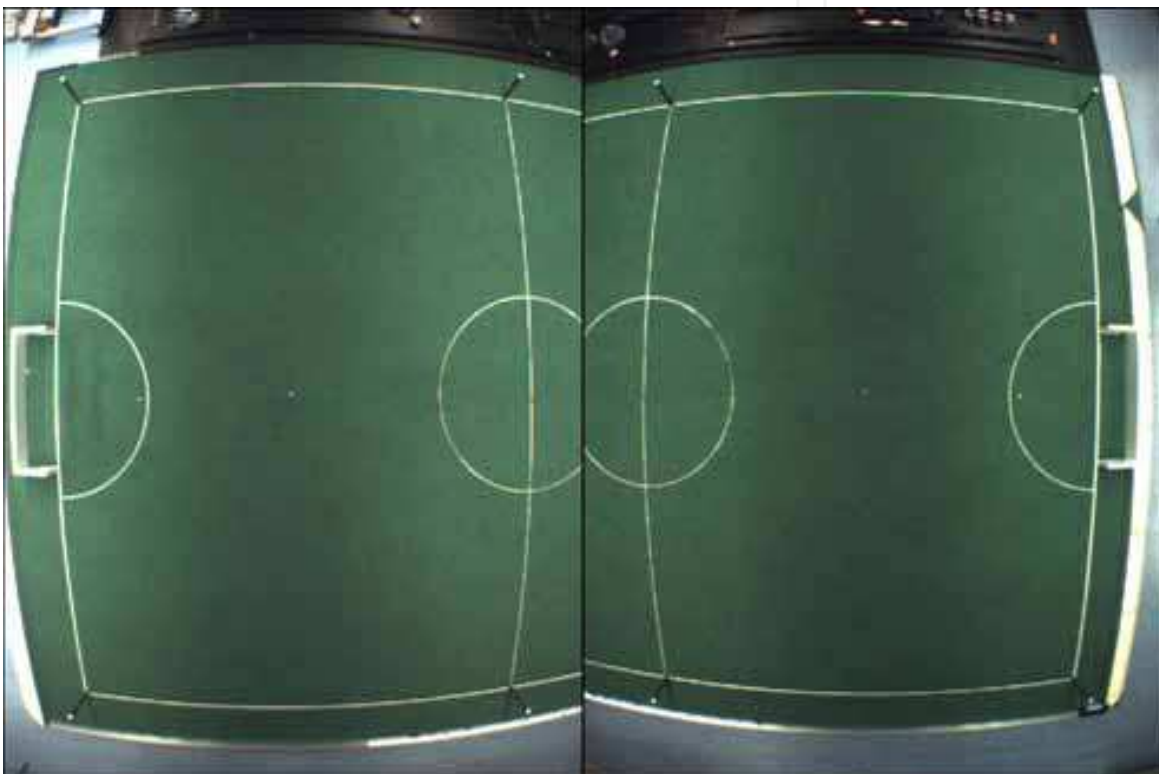


Fig. 10. Larger Robocup field without walls captured from two separate cameras. The poles placed in each corner of the field allow calibration of camera position.

7. Conclusion

The new calibration method requires negligible time to execute. Apart from the command to perform the calibration, it requires no user intervention, and is able to determine the model parameters in a fraction of a second. The model parameters are then used to automatically correct both the positions and orientations of the robots as determined from the distorted images. It is demonstrated that just capturing data from around the field is sufficient for correcting the whole playing area. The residual errors are significantly less than one pixel, and are limited by the resolution of the captured images.

The lateral position of the camera was able to be estimated to within 1 cm accuracy. The scaling effect of the parallax correction makes this error negligible. The height of the camera

is harder to measure accurately, because it is back-projecting the short height of the playing field walls. On two fields, it was within 8 cm, but on a third field the height was underestimated by 33 cm. However, even with this large error, the parallax error introduced in estimating the robot position less than one pixel anywhere on the playing field. Accurate height estimation requires good lighting, devoid of specular reflections near the walls, and for the walls to be in good condition.

The significant advantage of this calibration procedure over others described in the literature is that it is fully automated, and requires no additional setup or user intervention. While not quite fast enough to process every image, the procedure is sufficiently fast to perform recalibration even during set play (for example while preparing for a free kick) or a short timeout. It is also sufficiently accurate to support sub-pixel localisation and orientation.

8. Acknowledgements

This research was performed within the Advanced Robotics and Intelligent Control Centre (ARICC). The authors would like to acknowledge the financial support of ARICC and the School of Electrical and Electronic Engineering at Singapore Polytechnic.

9. References

- Bailey, D. & Sen Gupta, G. (2004). Error assessment of robot soccer imaging system, *Proceedings of Image and Vision Computing New Zealand (IVCNZ'04)*, pp 119-124, Akaroa, New Zealand, 21-23 November, 2004.
- Bailey, D. & Sen Gupta, G. (2008). Automatic estimation of camera position in robot soccer, *Proceedings of 2008 International Machine Vision and Image Processing Conference (IMVIP2008)*, pp 91-96, Portrush, Northern Ireland, 3-5 September, 2008.
- Bailey, D.G. (2002). A new approach to lens distortion correction, *Proceedings of Image and Vision Computing New Zealand 2002*, pp 59-64, Auckland, New Zealand, 26-28 November, 2002.
- Bailey, D.G. (2003). Sub-pixel estimation of local extrema, *Proceedings of Image and Vision Computing New Zealand 2003*, pp 414-419, Palmerston North, New Zealand, 26-28 November, 2003.
- Ball, D.M.; Wyeth, G.F. & Nuske, S. (2004). A global vision system for a robot soccer team, *Proceedings of 2004 Australasian Conference on Robotics and Automation*, pp 1-7, Canberra, 6-8 December, 2004.
- Baltes, J. (2000). Practical camera and colour calibration for large rooms, *Proceedings of RoboCup-99: Robot Soccer World Cup III*, pp 148-161, New York, USA, 2000.
- Basu, A. & Licardie, A. (1995). Alternative models for fish-eye lenses. *Pattern Recognition Letters*, vol. 16, no. 4, pp. 433-441.
- Brown, D.C. (1971). Close range camera calibration. *Photogrammetric Engineering*, vol. 37, no. 8, pp. 855-866.
- Egorova, A.; Simon, M.; Wiesel, F.; Gloye, A. & Rojas, R. (2005). Plug and play: fast automatic geometry and color calibration for cameras tracking robots, *Proceedings of RoboCup 2004: Robot Soccer World Cup VIII*, pp 394-401, Lisbon, Portugal, 2005.

- Fryer, J.G.; Clarke, T.A. & Chen, J. (1994). Lens distortion for simple C-mount lenses. *International Archives of Photogrammetry and Remote Sensing*, vol. 30, no. 5, pp. 97-101.
- Hartley, R. & Zisserman, A. (2000). *Multiple view geometry in computer vision*. Cambridge University Press.
- Heikkila, J. & Silven, O. (1996). Calibration procedure for short focal length off-the-shelf CCD cameras, *Proceedings of International Conference on Pattern Recognition*, pp 166-170, Vienna, Austria, 25-29 October, 1996.
- Klancar, G.; Kristan, M. & Karba, R. (2004). Wide-angle camera distortions and non-uniform illumination in mobile robot tracking. *Robotics and Autonomous Systems*, vol. 46, no. 2, pp. 125-133.
- Li, M. & Lavest, J.M. (1996). Some aspects of zoom lens camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 11, pp. 1105-1110.
- Messom, C.H. (1998). Robot soccer - sensing, planning, strategy and control, a distributed real time intelligent system approach, *Proceedings of Third International Symposium on Artificial Life and Robotics (AROB)*, pp 422-426, Oita, Japan, 19-21 January, 1998.
- Park, S.W. & Hong, K.S. (2001). Practical ways to calculate camera lens distortion for real-time camera calibration. *Pattern Recognition*, vol. 34, no. 6, pp. 1199-1206.
- Pers, J. & Kovacic, S. (2002). Nonparametric, model-based radial lens distortion correction using tilted camera assumption, *Proceedings of Computer Vision Winter Workshop 2002*, pp 286-295, Bad Aussee, Austria, 4-7 February, 2002.
- Press, W.H.; Flannery, B.P.; Teukolsky, S.A. & Vetterling, W.T. (1993). *Numerical recipes in C: the art of scientific computing*. Cambridge University Press.
- Sen Gupta, G.; Messom, C.H. & Sng, H.L. (2002). State transition based supervisory control for robot soccer system, *Proceedings of International Workshop on Electronic Design, Test and Applications (DELTA)*, pp 338-342, Christchurch, New Zealand, 29-31 January, 2002.
- Sen Gupta, G.; Messom, C.H. & Demidenko, S. (2004). State transition based (STB) role assignment and behaviour programming in collaborative robotics, *Proceedings of The Second International Conference on Autonomous Robots and Agents (ICARA 2004)*, pp 385-390, Palmerston North, New Zealand, 13-15 December, 2004.
- Stein, G.P. (1997). Lens distortion calibration using point correspondences, *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp 602-608, San Juan, Puerto Rico, 17-19 June, 1997.
- Sturm, P.F. & Maybank, S.J. (1999). On plane-based camera calibration: a general algorithm, singularities, applications, *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp 432-437, Fort Collins, Colorado, USA, 23-25 June, 1999.
- Tsai, R.Y. (1987). 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*, vol. 3, no. 4, pp. 323-344.
- Weiss, N. & Hildebrand, L. (2004). An exemplary robot soccer vision system, *Proceedings of CLAWAR/EURON Workshop on Robots in Entertainment, Leisure and Hobby*, pp, Vienna Austria, 2-4 December, 2004.
- Willson, R.G. & Shafer, S.A. (1994). What is the center of the image? *Journal of the Optical Society of America A*, vol. 11, no. 11, pp. 2946-2955.



Robot Soccer

Edited by Vladan Papi

ISBN 978-953-307-036-0

Hard cover, 348 pages

Publisher InTech

Published online 01, January, 2010

Published in print edition January, 2010

The idea of using soccer game for promoting science and technology of artificial intelligence and robotics was presented in the early 90s of the last century. Researchers in many different scientific fields all over the world recognized this idea as an inspiring challenge. Robot soccer research is interdisciplinary, complex, demanding but most of all, fun and motivational. Obtained knowledge and results of research can easily be transferred and applied to numerous applications and projects dealing with relating fields such as robotics, electronics, mechanical engineering, artificial intelligence, etc. As a consequence, we are witnesses of rapid advancement in this field with numerous robot soccer competitions and a vast number of teams and team members. The best illustration is numbers from the RoboCup 2009 world championship held in Graz, Austria which gathered around 2300 participants in over 400 teams from 44 nations. Attendance numbers at various robot soccer events show that interest in robot soccer goes beyond the academic and R&D community. Several experts have been invited to present state of the art in this growing area. It was impossible to cover all aspects of the research in detail but through the chapters of this book, various topics were elaborated. Among them are hardware architecture and controllers, software design, sensor and information fusion, reasoning and control, development of more robust and intelligent robot soccer strategies, AI-based paradigms, robot communication and simulations as well as some other issues such as educational aspect. Some strict partition of chapter in this book hasn't been done because areas of research are overlapping and interweaving. However, it can be said that chapters at the beginning are more system-oriented with wider scope of presented research while later chapters generally deal with some more particular aspects of robot soccer.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Donald G Bailey and Gourab Sen Gupta (2010). Automated Camera Calibration for Robot Soccer, Robot Soccer, Vladan Papi (Ed.), ISBN: 978-953-307-036-0, InTech, Available from:

<http://www.intechopen.com/books/robot-soccer/automated-camera-calibration-for-robot-soccer>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820

www.intechopen.com

Fax: +385 (51) 686 166
www.intechopen.com

Fax: +86-21-62489821

IntechOpen

IntechOpen

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen