Use of Stereo Vision Technology to Determine Position of Soccer Robots

Elmo B. de Faria[1], Juan F. B. Noguera[2] and Claudia A. Martins[3]

[1] University of Mato Grosso
IARA: Intelligence Artificial, Robotics and Automation Groups
Cuiaba, Brazil
Email: elmo {at} ic.ufmt.br

[2] Polytechnic University of Valencia
Valencia, Spain

[3] University of Mato Grosso
IARA: Intelligence Artificial, Robotics and Automation Groups
Cuiaba, Brazil

ABSTRACT— ***The propose of this study is to develop hardware and position algorithms completely integrated with the sensorial systems of robots to determine exactly the position of soccer robots in the playing field. These algorithms detail how to configure controllers to determine the movement of robots during the match.***
The study initially introduces some concepts of stereo reconstruction and triangulation in Computer Vision approaching, and provides an introduction to the calibration of cameras and types of triangulation. Finally is considers a structure to integrate camera systems with the movements of robots. An Experiment is shown which calculates the distance between robots and yours position in the real world.

Keywords— Soccer Robots, Computer Vision, Calibration and Triangulation.

1. INTRODUCTION

According to Truco and Verri [6] in the visual systems of animals, including man, the process of image formation begins with light rays coming from the outside world and impinging on the photoreceptors in the retina. The process of image formation in computer vision begins with de same light rays entering the camera through an angular aperture, and light intensities are registered (figure 1).

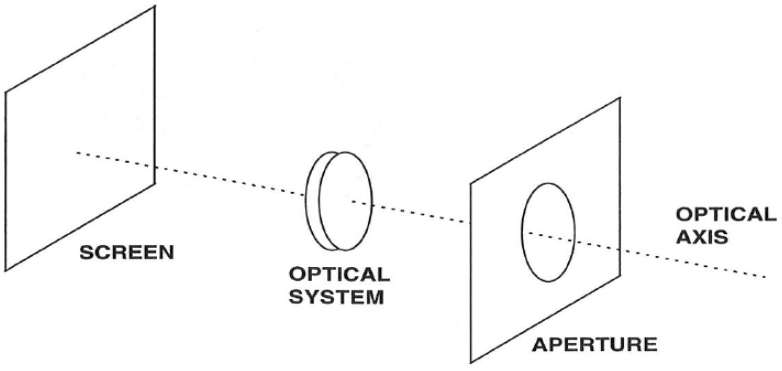


Figure 1- Elements of Imaging Device

1.1 - Camera Model

In the pinhole camera model [6][8], light enter from a scene or a distant object, but only a single ray enters from any particular point. In a physical pinhole camera, this point is then "projected" on to an imaging surface [2]. As a result, the image on this image plane (also called the projective plane) is always in focus, and the size of the image relative to the distant object is given by a single parameter of the camera: it's focal length. In the case our idealized pinhole camera, the

distance from the pinhole aperture to the screen is precisely the focal length. This is shown in Figure 2, where *f* is the focal length of the camera, *Z* is the distance from the camera to the object, *X* is the length of the object, and *x* is the object's image on the imaging plane. In the figure, we can see from the similar triangles that $-x/f = X/Z$, where

$$-x = f\frac{X}{Z} \qquad [1]$$

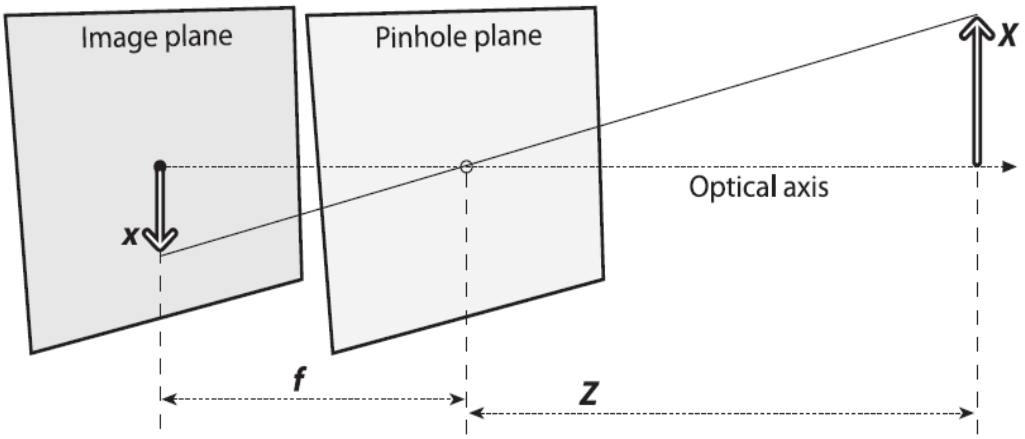


Figure 2- Pinhole Camera Model

The projection of the points in the physical world into the camera can be summarized by the following simple formula:

$$q = MQ \qquad [2]$$

where,

$$q = \begin{bmatrix} x \\ y \\ w \end{bmatrix}, M = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, Q = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \qquad [3]$$

Multiplying this out, we find that $w = Z$, and so, since the point q is in homogeneous coordinates, we should divide through by *w* (or *Z*) in order to return to equation [1]. The minus sign disappears because we are now looking at the non-inverted image on the projective plane in front of the pinhole rather than the inverted image on the projection screen behind the pinhole, (Figure 2).

1.2 - Camera Calibration

One of the main goals of computer vision is to understand the visible world by inferring 3D properties from 2D images [6]. In the context of stereo imagery, the first step that needs to be performed in the process of recovering 3D information from 2D images is known by the term calibration. Camera calibration is the process of computing the internal camera geometric and optical characteristics, and modelling the relationship between 2D images and the 3D world. Many types of calibration methods are presented in available literature.

Available literature suggests that they can be grouped into three main categories: traditional methods, self-calibration and active-motion based methods. The former method, the one that will be reviewed, is performed by observing a calibration object whose exact geometry in 3D space is known with precision. This method provided by Zhang [2] is of particular research interest, since it provides similar methodology to the one implemented by OpenCV platform, as well a common ground for data comparison.

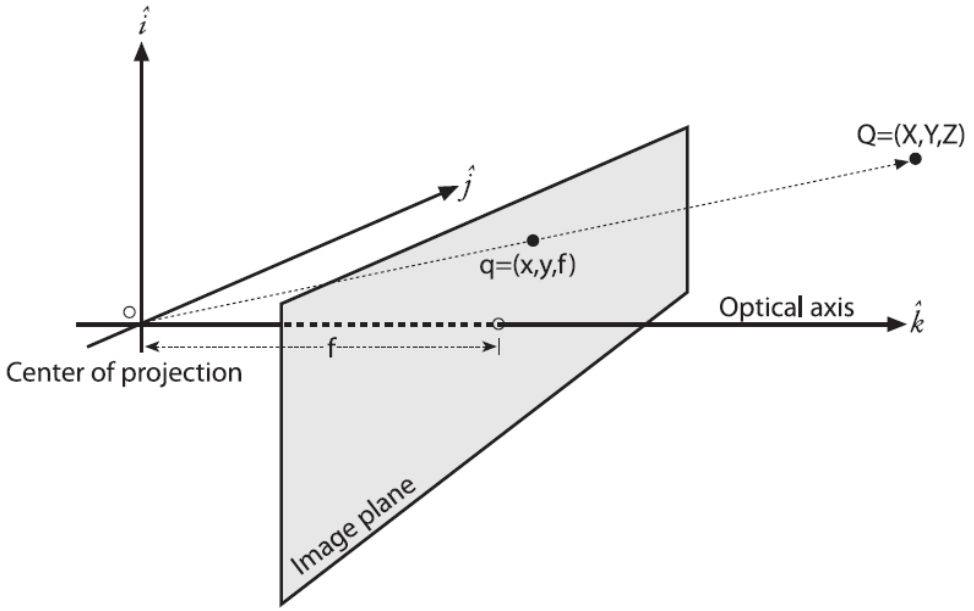


Figure 3 - Point Position in Real Space

The calibration object used in this study is a classic flat grid of alternating black and white squares that is usually called a "chessboard"(even though it need not have eight squares, or even an equal number of squares, in each direction), (Figure 4).

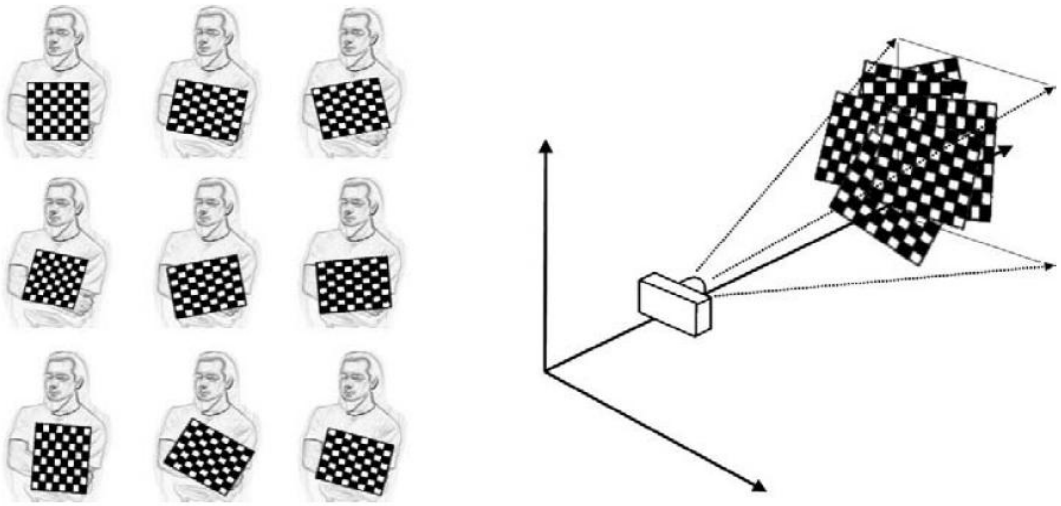


Figure 4- Chessboard Calibration Camera

Let us consider a 3D point in world coordinates $P = (X, Y, Z)^T$. We are assuming that the world reference system is known to readers. This 3D point may coincide with the center of projection of the camera (Although in general it does not need to). We shall let $Pc = (Xc, Yc, Zc)^T$ be the coordinates of the same point, this time in the camera reference frame, with $Zc > 0$ if the point is to be visible. The origin of the camera frame is its center of projection, and the Z axis is the optical axis. The extrinsic parameters of the camera are then the translation vector and the rotation matrix that effect the transformation from the world point to the same point in the frame of reference of the camera [12]:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + T \qquad [4]$$

where,

$$X_c = r_{11}X + r_{12}Y + r_{13}Z + T_x$$
$$Y_c = r_{21}X + r_{22}Y + r_{23}Z + T_x \quad [5]$$
$$Z_c = r_{31}X + r_{32}Y + r_{33}Z + T_x$$

Are the intrinsic parameters,

$$f_x = \frac{f}{s_x} \quad [6]$$

is the focal length in effective horizontal pixel size units

$\alpha = \frac{s_x}{s_y}$, the aspect ratio

(o_x , o_y) , the coordinates of the image center, and

k_1 , the radial distortion coefficient.

Combining the equations, we obtain:

$$x - o_x = -f_x \frac{r_{11}X + r_{12}Y + r_{13}Z + T_x}{r_{31}X + r_{32}Y + r_{33}Z}$$
$$y - o_y = -f_y \frac{r_{21}X + r_{22}Y + r_{23}Z + T_y}{r_{31}X + r_{32}Y + r_{33}Z} \quad [7]$$

$$f_y = \frac{f}{s_y} \quad [8]$$

Where in formula [8] assuming that the location of the image center (*ox* , *oy*) is known and that radial distortion can be ignored, the difficulty that presents itself to estimate *fx* , *α*, *R*, and *T* from image points $(xi,yi)^T$ which are the projection of N known world points $Pi=(Xi,Yi,Zi)^T$ obtained from the calibration pattern, in world coordinates.

This point in the image plane reference frame is:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{-fX_c}{s_x Z_c} + o_x \\ \frac{-fY_c}{s_y Z_c} + o_y \end{bmatrix} \quad [9]$$

1.3- Reconstruction by Triangulation

Simple reconstruction by triangulation is possible if the intrinsic and extrinsic parameters of the stereo system are known. Let us assume that we have a perfectly undistorted, aligned, and measured stereo rig as shown in Figure 5: two cameras whose image planes are exactly coplanar with each other, with exactly parallel optical axes (the optical axis is the ray from the center of projection) that are a known distance apart, and have equal focal lengths *fl = fr*. Also, let us assume for now that the principal points cx left and cx right have been calibrated to have the same pixel coordinates in their respective left and right images. A principal point is where the principal ray intersects the imaging plane. This intersection depends on the optical axis of the lens.

The image plane is rarely aligned exactly with the lens and so the center of the image is almost never exactly aligned with the principal point. With a perfectly undistorted aligned stereo rig and known correspondence, the depth *Z* can be found from the similar triangles; the principal rays of the images begin at the centers of projection *Ol* and *Or* and extend through the principal points of the two image planes at *cl* and *cr*

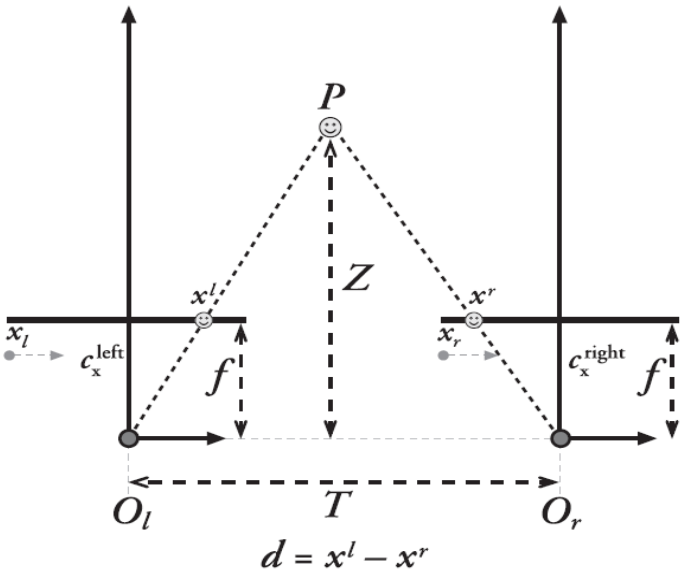


Figure 5 – Stereo Images Par

In this case, taking xl and *xr* to be the horizontal positions of the points in the left and right imager (respectively) the depth is inversely proportional to the disparity between these views, where the disparity is defined simply by $d = xl - xr$. This situation is shown in Figure 5, where we can easily derive the depth *Z* by using similar triangles. Referring to the figure 5, we have:

$$\frac{T-(x^l-x^r)}{Z-f}=\frac{T}{Z}\Rightarrow Z=\frac{fT}{x^l-x^r} \qquad [10]$$

Since depth is inversely proportional to disparity, there is obviously a nonlinear relationship between these two terms. When disparity is near 0, small disparity differences make for large depth differences. When disparity is large, small disparity differences do not change the depth by much. The consequence is that stereo vision systems have high depth resolution only for objects relatively near the camera. Figure 6 shows the 2D and 3D coordinate systems for stereo vision. Like in a right-handed coordinate system, if you point your right index finger in the direction of *X* and bend your right middle finger in the direction of *Y*, then your thumb will point in the direction of the principal ray. The left and right imager pixels have image origins at upper left in the image, and pixels are denoted by coordinates (*xl*, *yl*) and (*xr*, *yr*), respectively.

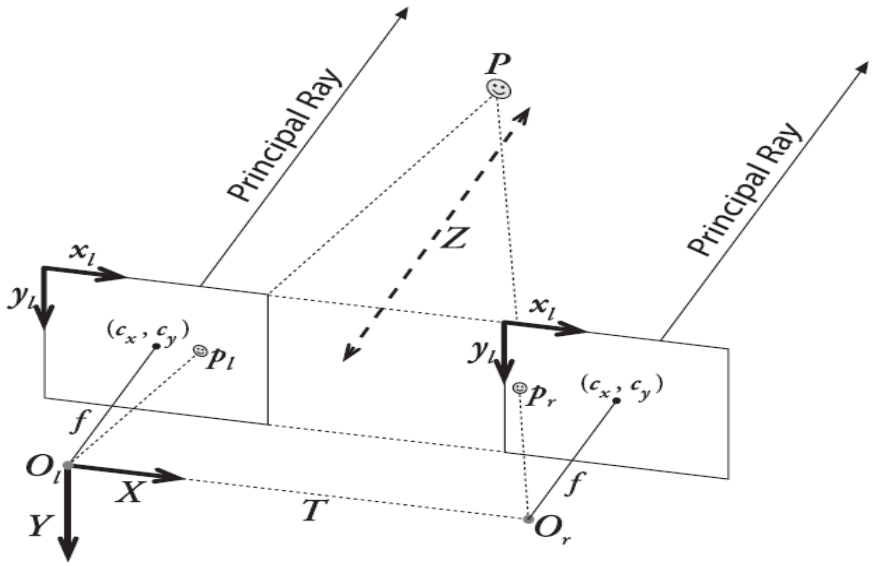


Figure 6 - Stereo Coordinate System

The center of projection is at *Ol* and *Or* and principal rays intersect the image plane at the principal point (*cx, cy*). After mathematical rectification, the cameras are row-aligned (coplanar and horizontally aligned), displaced from one another by *T*, and of the same focal length f. Mathematically it is possible to find image projections and distortion maps that will rectify the left and right images into a frontal parallel arrangement. When designing your stereo rig, it is best to arrange the cameras approximately frontal parallel and as close to horizontally aligned as possible. This physical alignment makes mathematical transformations more manageable. The mathematical alignment can produce extreme image distortions and so reduce or eliminate the stereo overlap area of the resulting images. Thus we, need synchronized cameras. This is a major problem for many cameras viewing in live images. With epipolar geometry, it is possible to have located corresponding points on the two or more stereo pairs of cameras. This geometry derives from Essential and Fundamental Matrix for stereo systems.

2. LABORATORY IMPLEMENTATION

This study presents two experiments in techniques to locate robots in the soccer field. The first presents results for two cameras in stereo systems to calculate the distance to the robots with a marker calibrator. In the second we propose a structure with six cameras for locating robots in the field during a game of soccer. These methods and materials are shown above, [9]. The robot used is the NAO standard platform league for the ROBOCUP, shown in figure 7.

NAO is a programmable, 58cm tall humanoid robot with the following key components:

- Body with 25 degrees of freedom (DOF) whose key elements are electric motors and actuators
- Sensor network, including 2 cameras, 4 microphones, sonar rangefinder, 2 IR emitters and receivers, 1 inertial board, 9 tactile sensors, and 8 pressure sensors
- Communication devices, voice synthesizer, LED lights, and 2 high-fidelity speakers
- Intel ATOM 1,6ghz CPU (located in the head) that runs a Linux kernel and supports Aldebaran's proprietary middleware (NAOqi)
- Second CPU (located in the torso)
- 27,6-watt-hour battery that provides NAO with 1.5 or more hours of autonomy,.

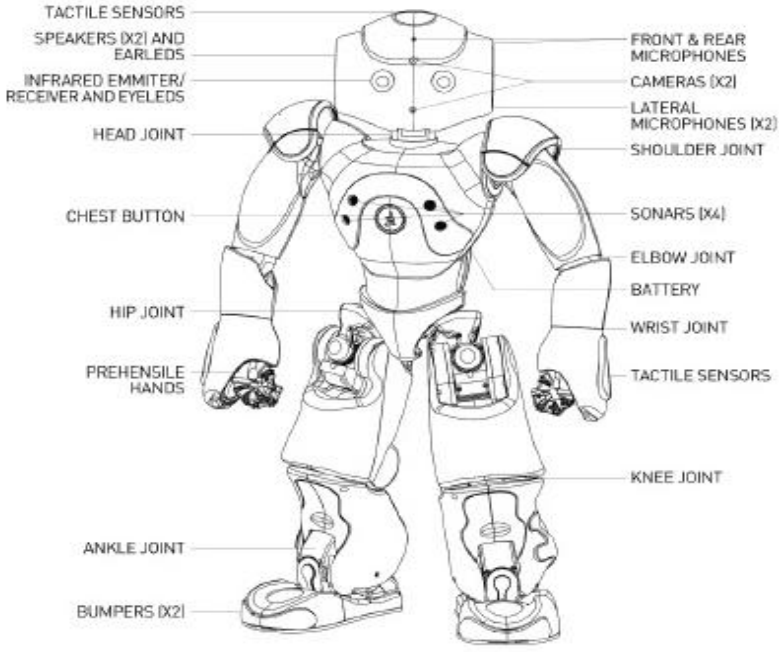


Figure 7 - NAO Soccer Robot Platform

The cameras for this experiment are the optitrack V100-R2, with specifications:

- Pixel Size : 6 µm × 6 µm
- Imager Size : 4.5 mm × 2.88 mm
- Imager Resolution : 640 × 480(0.3 MP)
- Frame Rate: 25, 50, 100 FPS
- Default Lens: 4.5mm F#1.6
    - Horizontal FOV: 46°
    - Vertical FOV: 35°

The calibration of cameras is done with OpenCV algorithms using the chessboard model shown in Figure 8. In this image OpenCV provides a convenient method for handling this common task. The function cvDrawChessboardCorners() draws the corners found by cvFindChessboardCorners() onto an image that you provide. If not all of the corners are found, the available corners will be represented as small red circles. If the entire pattern is found, then the corners will be painted in different colours (each row will have its own colour) with connected bylines representing the identified corner order.

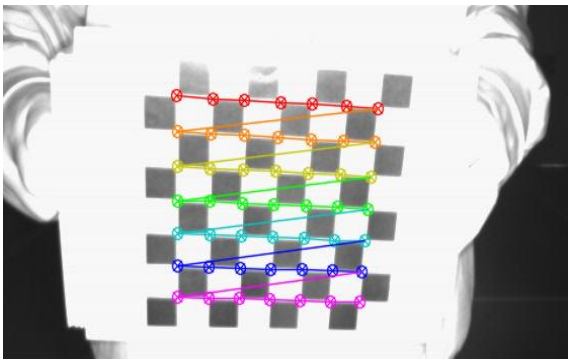


Figure 8 - Camera Chessboard Calibration

After the calibration process, other functions can use like, Stereo Correspondence and Stereo Calibration, both of which are tested in this project. A general diagram of the robotic system and the calibrated cameras is shown in figure 9. In this instance, only one robot is used to calculate the distance of the marker for future estimates of real positions in the soccer field. The robots have a constant height; in this case the *Z* coordinate for word systems is 60cm. To determine different positions in this axis, the bar marker is used and different distances are obtained. The positions and height of cameras up the center of the soccer field (reference *XYZ* in real world terms) are known. The systems propose working in real-time. This means that, when a robot walks the distance measured changes and the new position is calculated. The data are used to correct de position of the robot and to validate the embedded algorithms. These algorithms are often is a strategic player in the real game as simulated on desktops and in exhaustive tests.

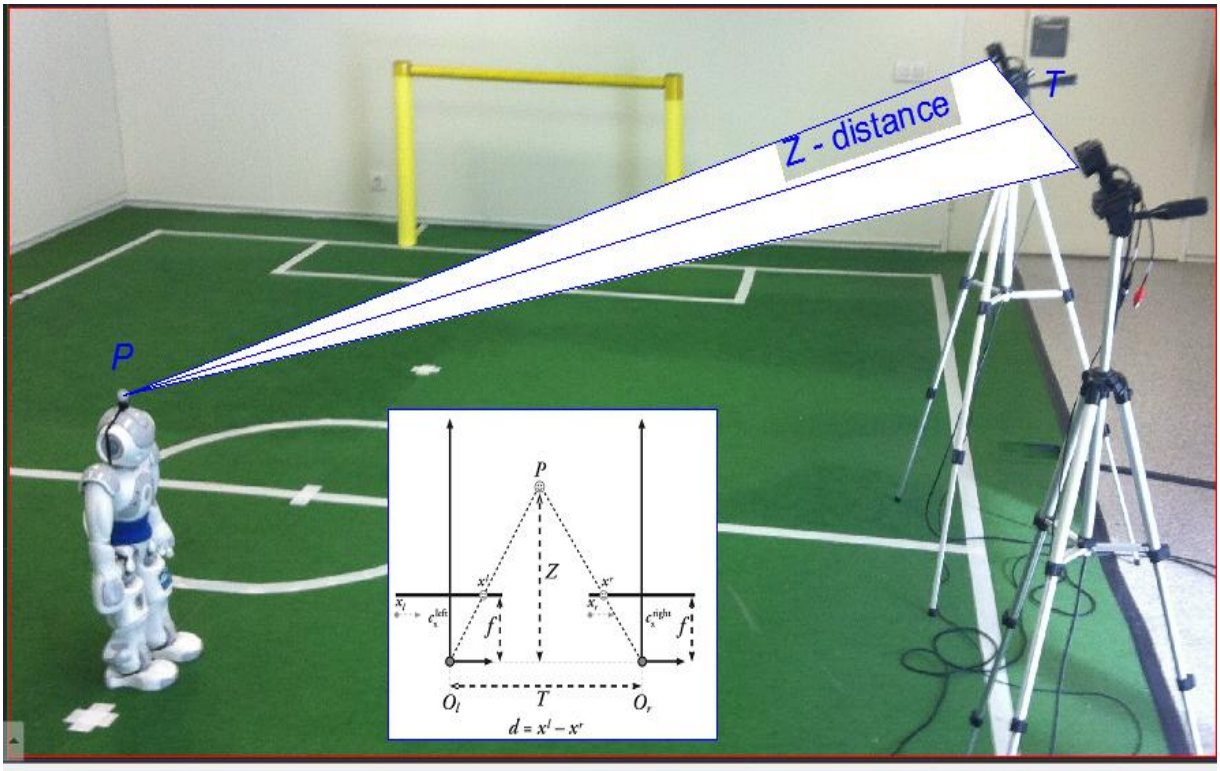


Figure 9 - General Schematic

Figure 10 illustrates in detail the image and object identification of a marker on the head of a robot. This marker is a reflective infrared object. The cameras detect this marker and convert it into XY positions for Left and Right images. Objects witch are not reflective are not detected and are discarded. Note that for both cameras 0 and 1 only de marker is detected other elements in view are ignored.

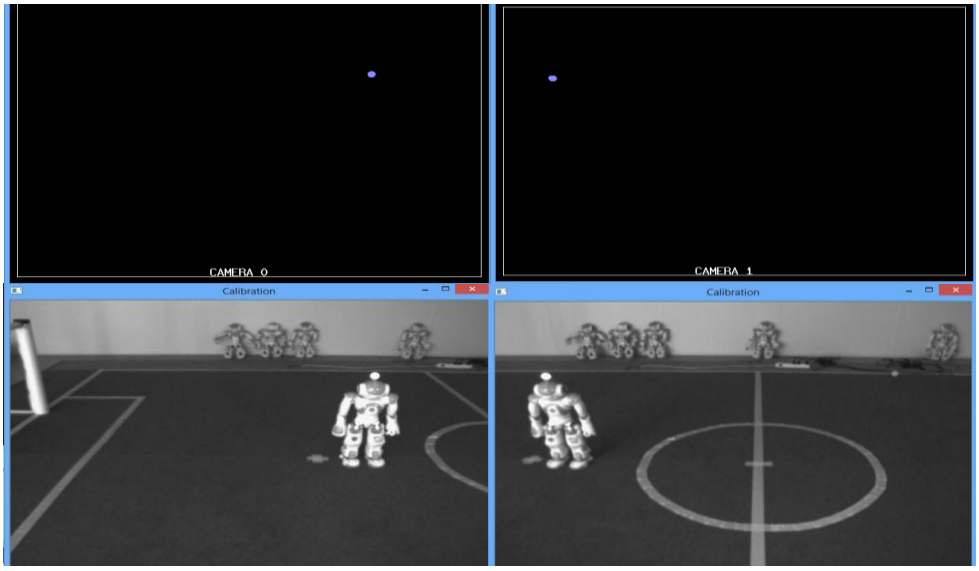


Figure 10 – Robot Image in Camera Identification

In order to estimate the distance in different height levels, a marker calibrator is used. Figure 11 shows the marker and the distance Z0, Z1, Z2 and Z3 for the stereo system.

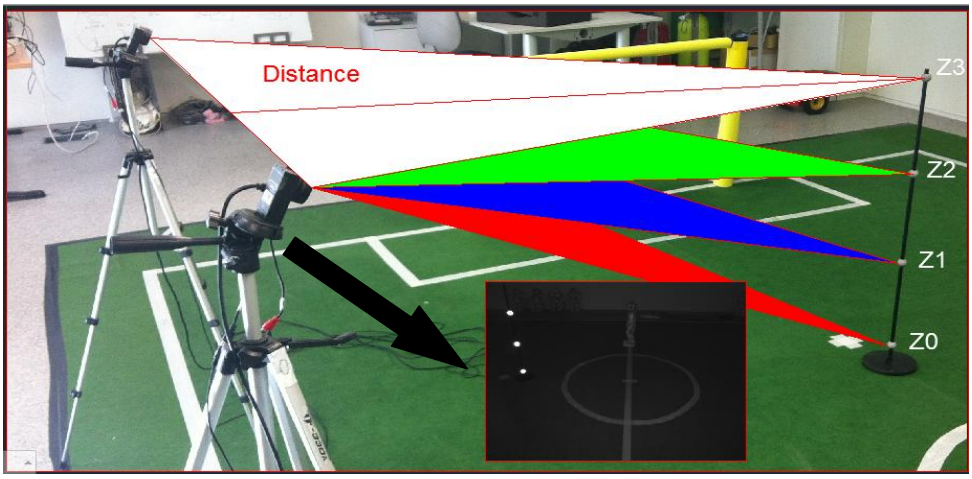
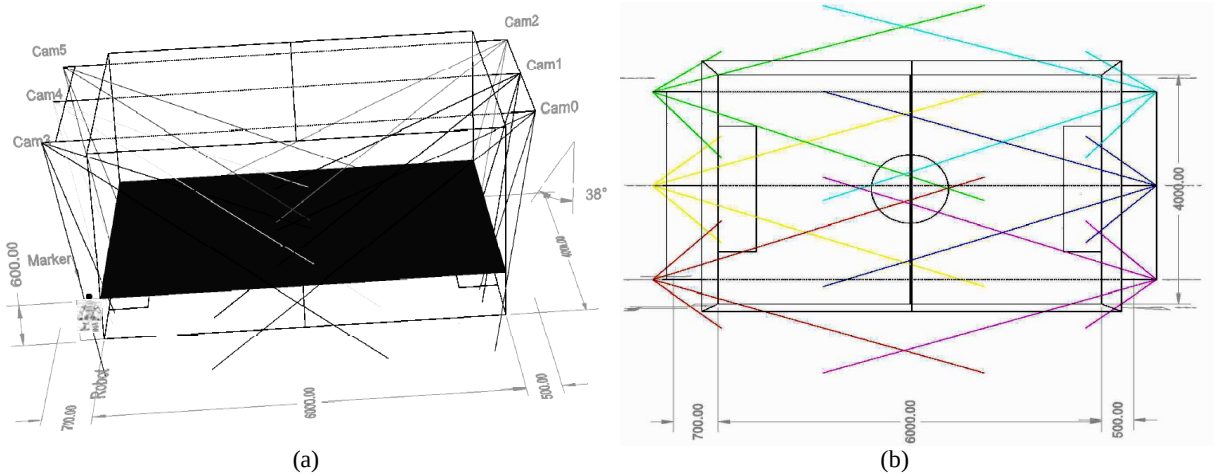


Figure 11 – Calculation of Distance and 3D Position

In another test, six cameras are used to estimate the position of robots in a simulated game of soccer. For this situation there is a new arrangement as shown in Figure 12.



(a) (b)

Figure 12 – (a) Reference Position System for Two Teams in a Game of Robot Soccer. (b) Top View.

All cameras are synchronized and a computer identifies and triangulates the positions of two teams of soccer robots (6 robots). This information is obtained and filed in a data file system. These files stay to be read for the robots team, [10][11].

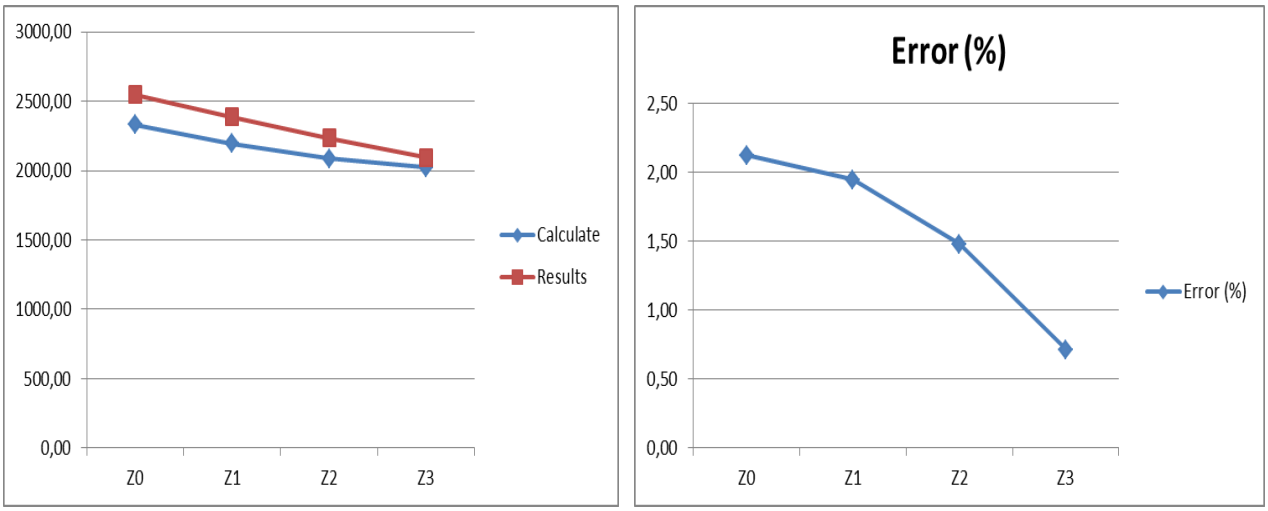
3. RESULTS

Table 1, shows the estimated distance to four markers for the arrangement shown in Figure 11. The column "Calculated" shows the values obtained by calculating distance with triangle algebra. The column "Results" shows the values obtained directly from the reconstructed algorithms for triangulation. The percentage error is shown in the last column of the table.

Table 1 – Distance Results

Distance	X cam0	Y cam0	X cam1	Y Cam1	Calculated	Results	Error (%)
Z0	77.02	245.02	582.18	261.52	2332.38	2545.00	2.13
Z1	60.76	172.62	599.09	188.47	2193.17	2388.00	1.95
Z2	42.59	90.88	617.42	105.87	2088.06	2236.22	1.48
Z3	22.60	0.42	636,71	10.98	2022.37	2093.76	0.71

Figure 13(a) shows the results of the calculations of distance. The data source is shown in Figure 14. The error is around 2.5% for this test. In a system like that in Figure 12, where six cameras are used, many cameras provide many different positions for markers. In this case it is possible to do calculate an average value for data and reduce the error. Figure 14 shows the software development for this project. The platform is Windows, and Visual Studio, C++ is the language used.



(a) (b)

Figure 13 – (a) Distance From Position of Markers ;(b) Errors in Calculated Distances

4. CONCLUSIONS

The results obtained from the laboratory experiments show that this system is able to determine the position of robots in a soccer field. It is clear that position is more precise for robots near the center of cameras. Like in equation [8] the relation of disparity *(d = xl – xr)* is inversely proportional to depth. There is clearly a non-linear relationship between these two terms. When disparity is near 0, small disparity differences make for large differences in depth. When disparity is large, small disparity differences much. The consequence is that stereo vision systems have high depth resolution only for objects relatively near to the cameras. Figure 13-(b) clearly demonstrates that errors are minor for markers near to cameras, and major for markers far from of cameras. This is less of a problem when many cameras are used because an average result can be used. Another process under investigation is the use of Neural Networks to estimate the 3D real position for robots in the soccer field.

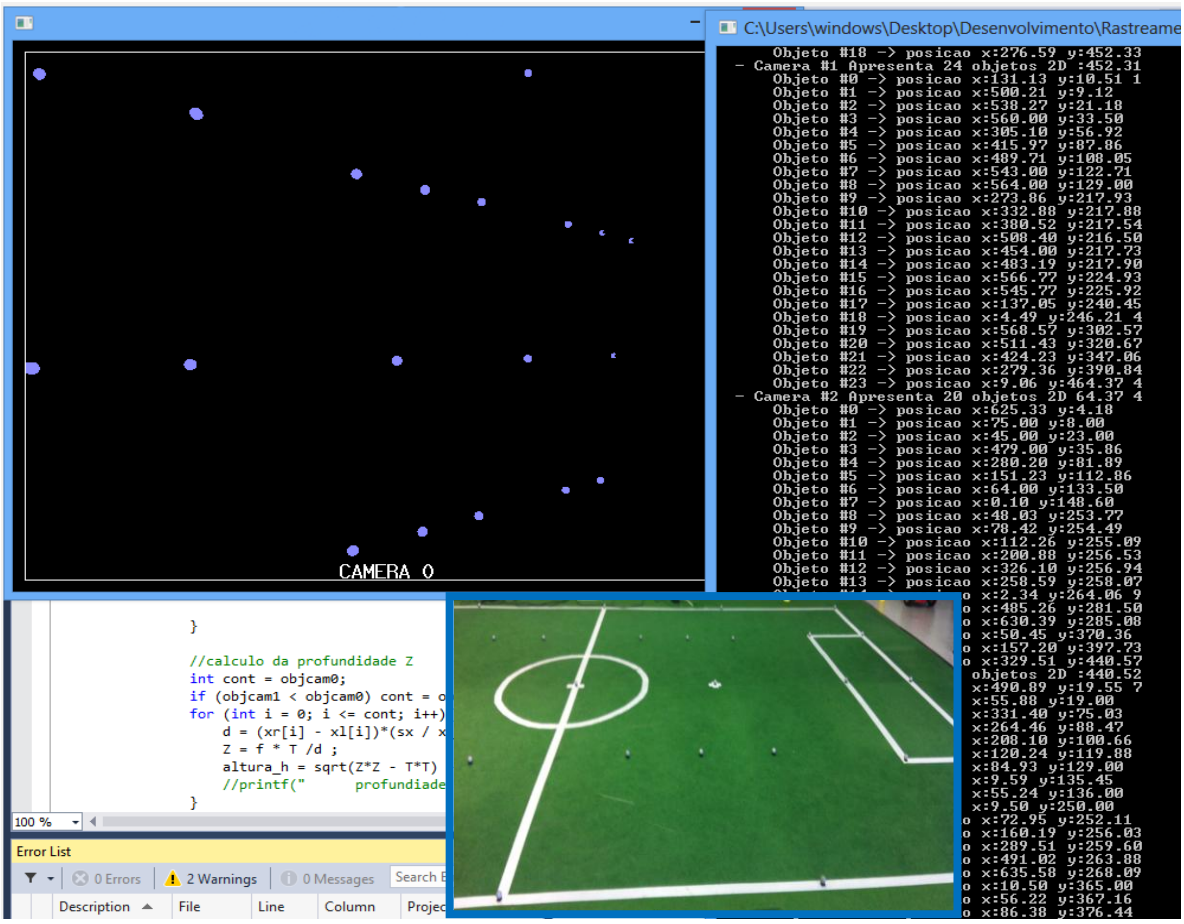


Figure 14- Software for Location Markers with 6 Cams

5. REFERENCES

[1] Okko H. Bosgra, Huibert Kwakernaak and Gjerrit Meinsma, "Design Methods for Control Systems", 2009.
[2] G. Bradski and A. Kaehler. Learning OpenCV. O'Reilly Media, Inc., Sebastopol, CA, _rst edition edition, 2008.
[3] G. Burdea and P. Coi_et. Virtual Reality Technology. John Wiley & Sons, Inc, Hoboken, New Jersey, second edition edition, 2003.
[4] K. Kirby, J. Forlizzi, R. Simmons. Affective social robots. Robotics and Autonomous Systems, 2010.
[5] J. Shi and C. Tomasi. Good Features To Track. Technical report, IEEE Conference on Computer Vision and Pattern Recognition, June 1994.
[6] E. Trucco and A. Verri. Introductory Techniques for 3-D Computer Vision. Prentice Hall, Inc., Upper Saddle River, New Jersey, edition, 1998.
[7] R. I. Hartley and A. Zisserman, _Multiple View Geometry in Computer Vision._ Cambridge University Press, 2004.
[8] S. T. Bernard, _A stochastic approah to stereo vision,_ Philadelphia, USA, 1986.
[9] R. S. Nourbakhsh and I. R., Introduction to Autonomous Mobile Robots, in Bradford Company, MA, USA, 2004.
[10] R. M. H.D. Burkhard, D.Duhaut, M.Fujita, P.Lima and R. Rojas, _The road to robocup 2050,_ Robotics & Automation Magazine, IEEE, 9(2):31-38.
[11] M. Shibata and N. Kobayashi, _Image-based visual tracking for moving targets with active stereo vision robot, in SICE-ICASE, 2006. International Joint Conference, 2006, pp. 5329_5334.
[12] J. K. Yi Ma, Stefano Soatto and S. S. Sastry, _An Invitation to 3-D Vision, from images to geometric models._ in Springer, 2006.
[13] P. M. David Gallup, Jan-Michael Frahm and M. Pollefeys, Variable baseline/resolution stereo,_ 2008.
[14] S. M. Nixon, Feature Extraction in Computer Vision and Image Processing.
[15] R. I. Hartley., _Theory and Practice of Projective Rectification._ International Journal of Computer Vision, 35(2): 115-127, 1999.
[16] C. . Z. Z. Loop, _Computing Rectifying Homographies for Stereo Vision,_ in IEEE Conference on Computer Vision and Pattern Recognition, Colorado, USA, 1999.
[17] OpenCV. (Open Source Computer Vision Library). [Online]. Available: www.opencv.org
[18] Armangue, X. e Salvi, J. (2003). Overall view regarding fundamental matrix estimation. Image and Vision Computing, 21:205–220.