Real-time Teaching of Industrial Robots Using a Synchronised Stereoscopic Vision System

Paulo Malheiros, Paulo Costa, António Paulo Moreira and José Carlos Lopes

***Abstract*— In this paper a method to control and teach industrial robots in real-time by human demonstration is presented. This system uses high-intensity visual markers that make it sufficiently robust for industrial environments not requiring special lighting. An automated camera calibration method was implemented which enables any non-skilled user a quick and easy configuration of the system.**

The teaching of the robot is achieved by recording the detected points and replaying them to the manipulator, therefore this is a "teaching-by-showing" method.

This system can work with any kind of industrial manipulator capable of receiving remote positioning commands.

I. INTRODUCTION

Industrial robots are programmable multifunctional mechanical devices designed to move material, parts, tools, or specialised devices through variable programmed motions to perform a variety of tasks. Robots are generally used to perform unsafe, hazardous, highly repetitive, and unpleasant tasks. Most robots are set up for an operation by the teach-and-repeat technique. In this mode, a trained operator (programmer) typically uses a portable control device (a teach pendant) to teach a robot its task manually [1].

The required flexibility in manufacturing implies that both machine control and equipment interfaces need to be easy to change for new application scenarios [2]. Robot programming is not easy for novice operators and the cost of training them is often unaffordable especially for small companies. Thus low-cost and labour-saving robot teaching is greatly to be desired [3].

Consequently, new methodologies for programming robots quickly, safely and intuitively are desired. To this purpose, interactive programming interfaces that allow non-skilled users to program robots have been developed over the years. These developments require higher levels of abstraction and in some way tend to lead to machines that understand human-like instructions [4].

A simple real-time "teaching-by-showing" method has been developed for conventional industrial robots, shown in Fig. 1. This application focuses on improving efficiency by enhancing human-robot interaction for task generation. Although several developments have been done over the years, this technology is still far from industrial applications since previous approaches require special equipment (data gloves, position sensors, etc.) and specific environment condition (controlled lighting, large teaching equipment, etc.).

Paulo Malheiros is a PhD Candidate, Paulo Costa and António Paulo Moreira are Assistant Professors with the Department of Electrical and Computer Engineering, Faculty of Engineering of the University of Porto, Rua Dr. Roberto Frias s/n, 4200 465 Porto, Portugal paulo.malheiros@fe.up.pt

José Carlos Lopes is an Associate Professor with the Department of Chemical Engineering, Faculty of Engineering of the University of Porto, Rua Dr. Roberto Frias s/n, 4200 465 Porto, Portugal

Fig. 1. Teaching system with industrial manipulator

The presented method uses image processing synchronised with high intensity markers (e.g. LEDs). These markers are so bright that their light outshines the surrounding environment making them easy to detect using regular cameras. The synchronisation with the cameras allows the markers to be only active during the image acquisition (with a few milliseconds duration) which makes its apparent light very dim not interfering with human vision.

The markers are attached to the user's tools (e.g. paint spray gun) and using a stereoscopic vision system, shown in Fig. 2, it is possible to register the path in space and make the robot manipulator mimic the same movements. Three or more markers allow the detection of the six degrees of freedom of the tool so all the robot's axis are correctly controlled.

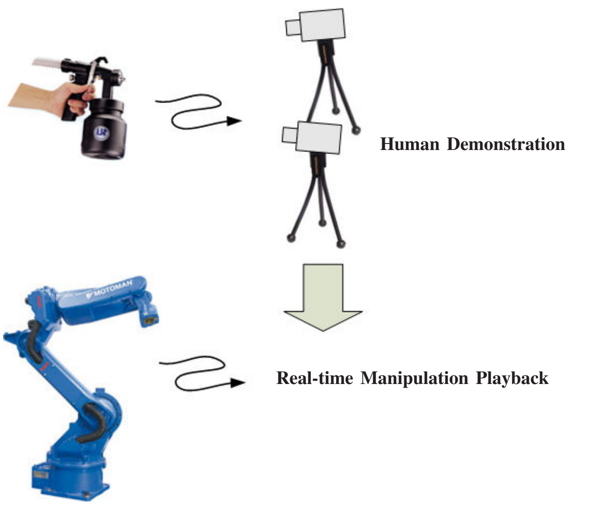


Fig. 2. Manipulator moving simultaneously with user's movement

The system can work in real-time because the markers are easily identified and classified, therefore the user can instantly see the resulting path in the manipulator. The teaching process is achieved easily by replaying the previously recorded sequence.

This system is ideal to work as a remote control which can be useful in hazardous and unsafe conditions, like controlling the space station robotic arm or moving radioactive material.

II. TEACHING SYSTEM

For the developed system we set the following assumptions:

- Uses common and non-expensive equipment.
- Non-obtrusive sensors so that users do their natural movements.
- Works in any industrial environment.
- Easy implementation for non-skilled users.

This system consists of two Firewire industrial cameras, one laptop, a synchronisation system and a set of LEDs. The cameras are two FireWire CCD Bayer industrial cameras with 640x480 resolution and external trigger that allows the synchronisation of their acquisition through an input clock. This synchronisation signal is controlled by a generic board with an 8-bit microcontroller which also triggers the LED markers. The synchronisation for the markers can be a radio or infrared signal thus reducing the cabling which can cause inconvenience with the user. In Fig. 2 is represented the use of this system with an industrial manipulator in a painting application.

An image processing application was developed in Object Pascal using Lazarus and using the 5dpo Component Library [6] which is an Open Source set of components developed by the authors for image acquisition. The application can also position the markers in a OpenGL 3D virtual world using GLScene components.

The markers are standard high-power LEDs which have approximately 4 Watt and come in several colours. The use of different colours allows the immediate classification of several markers by coloured image processing application. In Fig. 3 is represented an high-powered led easily detected by the camera which reduces the time needed to process the image, this is essential in real-time applications. Even reducing the camera aperture the marker is easily detectable while the surrounding environment fades away.

Having detected the the tool position this system can order any kind of industrial manipulator (e.g. XY Table) with a remote control interface. A Motoman HP6 Series industrial robot was used together with a Motoman NX100 Controller for this paper, shown in Fig. 1. The controller has a *Remote* mode that receives the control instructions sent by the PC through an Ethernet connection.

This system is easily installed and with little time and effort is fully operational. For this we need:

1) Install the cameras and the markers in the control tool.
2) Calibrate the cameras stereoscopy using the previous tool.
3) Fix the system scale and distortion using known world points.

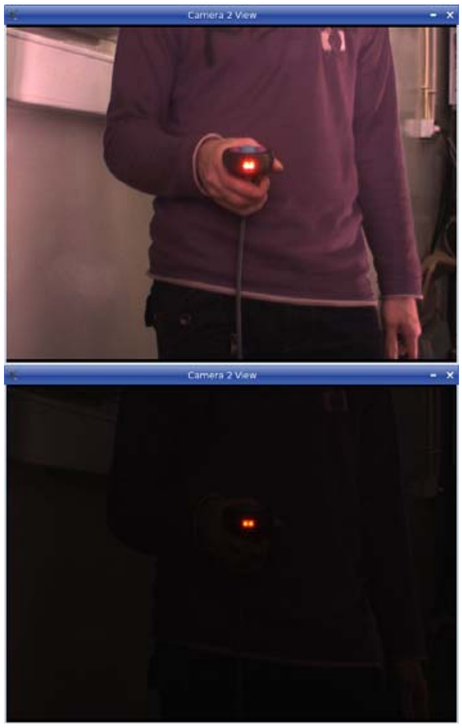


Fig. 3. LED detection in an uncontrolled environment. Only marker visible when camera's aperture is closed

4) Translate and Rotate the world coordinates to make them equal to the robot coordinates.
5) Reconstruct the tool position from the measured markers and playback in the robot.

In five steps we implement a high-precision and real-time teaching system for industrial robots.

III. CAMERAS CALIBRATION

Once the cameras are positioned in a new installation there is the need to mathematically model the imaging system. This is done using the *Fundamental Matrix* that encapsulates the epipolar geometry of the imaging configuration. We used the Normalised Eight-point Algorithm which is the simplest method of computing the fundamental matrix, involving no more than the construction and (least-squares) solution of a set of linear equations [7]. Finally it's possible to estimate the 3D position of any given point in a pair of images having the fundamental matrix determined.

We illustrate the steps taken to test the principle and possibilities of this technology. It was demonstrated using a single marker to control the position of the manipulator. Position and orientation are to be tested in the future as well as precision studies.

A. Notation

In this paper the following notation is followed:

- The vectors are represented by lowercase bold letters (e.g. $\mathbf{u}$) when it refers to 2D space in homogeneous coordinates and are thought of as being column vectors unless explicitly transposed. Uppercase bold letters when it refers to 3D space (e.g. $\mathbf{T}$) homogeneous coordinates.
- The matrices are represented by uppercase fixed size letters (e.g. A).

B. Camera Intrinsic Parameters

The first step was to determine and correct the cameras barrel distortion using a known pattern (Fig. 4). Once the second order barrel distortion and centre of the image were determined all the images were treated has not having any kind of radial distortion in the following stages.

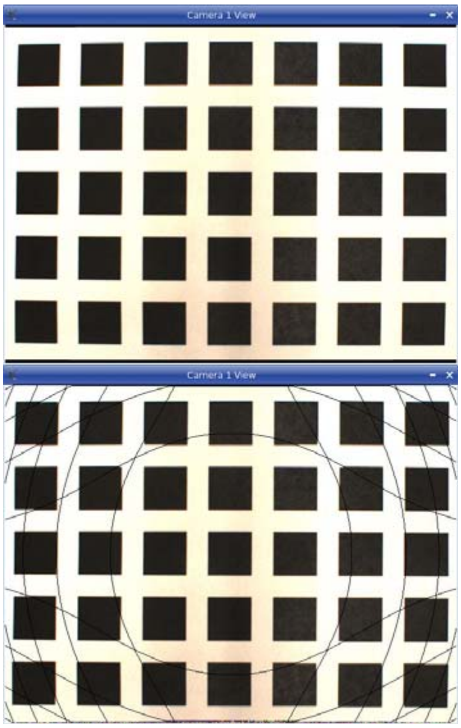


Fig. 4. Pattern used for determining the intrinsic parameters. Sample barrel distortion compensation

The inverse radial distortion function is the mapping from the distorted point $\mathbf{p}_d$ to the undistorted point $\mathbf{p}_u$. It can be concluded from the location of the point of sharp projection $\mathbf{p}$ that the radial distortion increases with the radius r. Thus, the inverse radial distortion function $f(r_d)$ can be approximated and parametrised by the following Taylor expansion [8]:

$$r_u = r_d + r_d \sum_{i=0}^{\infty} \kappa_i r_d^{i-1} \tag{1}$$

with

$$r_u = \sqrt{x_u^2 + y_u^2} \quad \text{and} \quad r_d = \sqrt{x_d^2 + y_d^2}$$

it follows that

$$x_u = x_d + x_d \sum_{i=0}^{\infty} \kappa_i r_d^{i-1} \tag{2}$$

$$y_u = y_d + y_d \sum_{i=0}^{\infty} \kappa_i r_d^{i-1} \tag{3}$$

For these tests only κ_3 was taken into account since practical tests have shown that this makes a good radial correction. The parameter κ_3 has the dominant influence on the kind of radial lens distortion. If $\kappa_3 > 0$, a barrel distortion and if $\kappa_3 < 0$, a pincushion distortion is compensated by f(rd). Thus, we simplify Eq. 2 and Eq. 3 to:

$$x_u = x_d + x_d \kappa_3 r_d^2 \tag{4}$$

$$y_u = y_d + y_d \kappa_3 r_d^2 \tag{5}$$

Finally the intrinsic matrix K (also called *camera calibration matrix*) maps the normalised image coordinates to the retinal image coordinates [10].

$$K = \begin{bmatrix} f k_u & 0 & u_0 \\ 0 & f k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

Using the same pattern we were able to determine the focal length f. The horizontal and vertical scale factors, k_u and k_v, are the inverse of the size of the CCD pixel. u_0 and v_0 the intersection between the optical axis and the retinal plane.

C. Normalised Eight-point Algorithm

Binocular vision consists in using two cameras to view a point in the space $\mathbf{U}$. Fig. 5 shows the optical centers $\mathbf{C}_1$ and $\mathbf{C}_2$, $\mathbf{u}$ and $\mathbf{u}'$ are the images of $\mathbf{U}$ on the retinal planes Π_1 and Π_2.

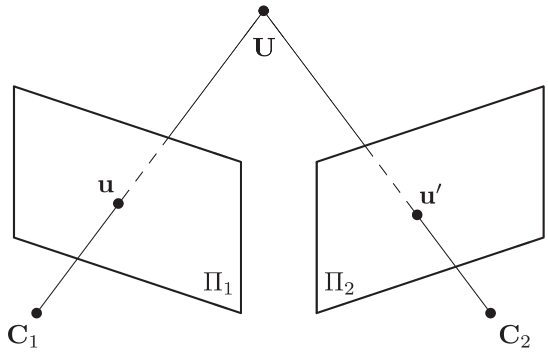


Fig. 5. Binocular vision

The fundamental matrix is a 3×3 matrix which relates corresponding points in stereo images defined by the equation

$$\mathbf{u}'^T F \mathbf{u} = 0 \tag{7}$$

for any pair of matching points $\mathbf{u}' \leftrightarrow \mathbf{u}$ in two images. Given sufficiently many point matches $\mathbf{u}'_i \leftrightarrow \mathbf{u}_i$ (at least eight) this Equation 7 can be used to compute the unknown matrix F [9]. Specifically, the equation corresponding to a pair of points $\mathbf{u} = (u, v, 1)^T$ and $\mathbf{u}' = (u', v', 1)^T$ in will be

$$\begin{aligned} &uu'F_{11} + uv'F_{21} + uF_{31} + vu'F_{12} + vv'F_{22} + \\ &vF_{32} + u'F_{13} + v'F_{23} + F_{33} = 0 \end{aligned} \tag{8}$$

The row of the equation matrix may be represented as a vector

$$(uu', uv', u, vu', vv', v, u', v', 1) \tag{9}$$

From all the point matches, we obtain a set of linear equations of the form

$$A\mathbf{f} = 0 \tag{10}$$

where $\mathbf{f}$ is a nine-vector containing the entries of the matrix F, and A is the equation matrix. The fundamental matrix F, and hence the solution vector $\mathbf{f}$ is defined only up to an unknown scale.

The key to success with the eight-point algorithm is proper careful normalisation of the input data before constructing the equations to solve. A simple transformation (translation and scaling) of the points in the image before formulating the linear equations leads to an enormous improvement in the conditioning of the problem and hence in the stability of the result. The added complexity of the algorithm necessary to do this transformation is insignificant.

The suggested normalisation is a translation and scaling of each image so that the centroid of the reference points is at the origin of the coordinates and the RMS distance of the points from the origin is equal to $\sqrt{2}$ [7].

The fundamental matrix F is then obtained denormalising the previous result.

IV. 3D Reconstruction

Any camera can be now positioned in the world using their rotation matrix R and translation vector $\mathbf{T}$ determined from the fundamental matrix. The main property of the camera model is that the relationship between the world coordinates and the pixel coordinates is linear projective. This relationship remains linear regardless the choice of both coordinates [10]. Thus, the 3D point $\mathbf{U}$ and its camera projection $\mathbf{u}$ are related by

$$s\mathbf{u} = P\mathbf{U} \tag{11}$$

where s is a scale factor called *depth* and P is a 3×4 matrix called the *perspective projection matrix*.

To simplify we can consider the first camera as being in the origin then $(R, \mathbf{T})$ is the displacement from the first camera to the second, under the pinhole model, we have the following two equations

$$\begin{aligned} s\mathbf{u} &= K[I \mid \mathbf{0}]\mathbf{U} \\ s'\mathbf{u}' &= K'[R \mid \mathbf{T}]\mathbf{U}' \end{aligned} \tag{12}$$

where K and K' are the intrinsic, or calibration matrices of both cameras [11] determined previously. A 3D point is the intersection of two rays passing through the optical centers of each camera, and corresponding image points $(\mathbf{u}', \mathbf{u})$. In Fig. 6 is represented a set of reconstructed points used during the calibration. It represents the tool movement in space which is basically a random movement to result in random points.

The determination of the scale factor s was done placing a marker in the industrial manipulator and making this move a precise distance. The relation between the real distance and the measured distance gives this scale factor.

A 3D point is the intersection of two rays passing through the optical centers $\mathbf{C}_1$ and $\mathbf{C}_2$, and corresponding image points $(\mathbf{u}, \mathbf{u}')$. The three points $\mathbf{u}$, $\mathbf{u}'$ and $\mathbf{U}$ form a triangle. Thus, the final task is only to resolve reconstruct the 3D point through triangulation.

V. PATH DETECTION

The demonstration of the technology of controlling and teaching industrial manipulators was the main purpose of the paper. The tests were made using only a single marker so the presented results are only of three degrees of freedom.

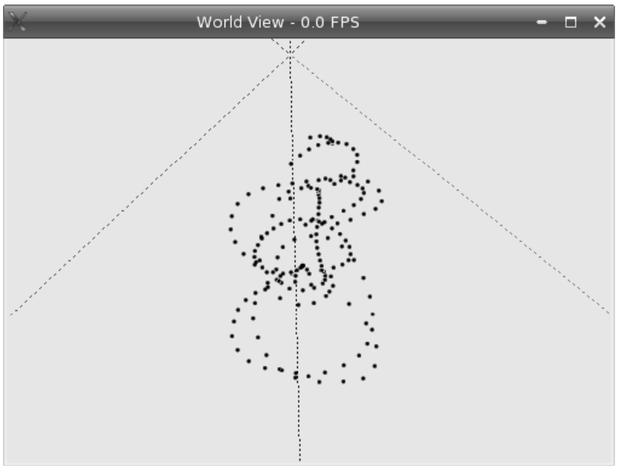


Fig. 6. Reconstructed calibration points

A. Real-time Playback

The reconstructed system are centred on the principal camera, in Fig. 6 the origin is away from calibration points which is the exact position of the camera looking at the points. Ideally the tool coordinates should be the same as the ones in the manipulator so this system can translate and rotate its coordinates.

The coordinates translation is done showing a central point with the tool, followed by the rotation of the axis showing two points placed in two of the robots axis. Fig. 7 shows the previous calibration points with the manipulator coordinates, the calibration was done in the middle of our working environment.

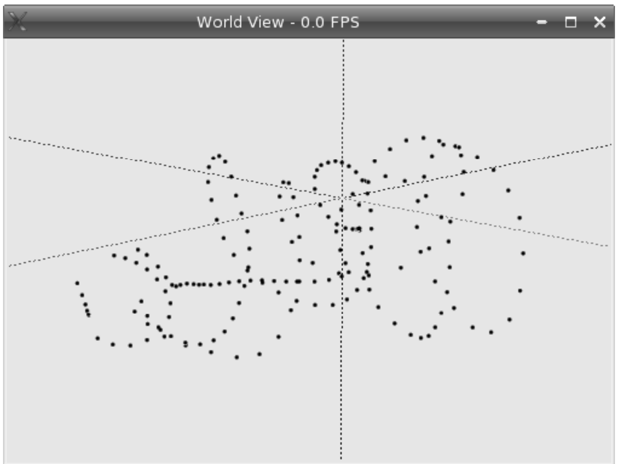


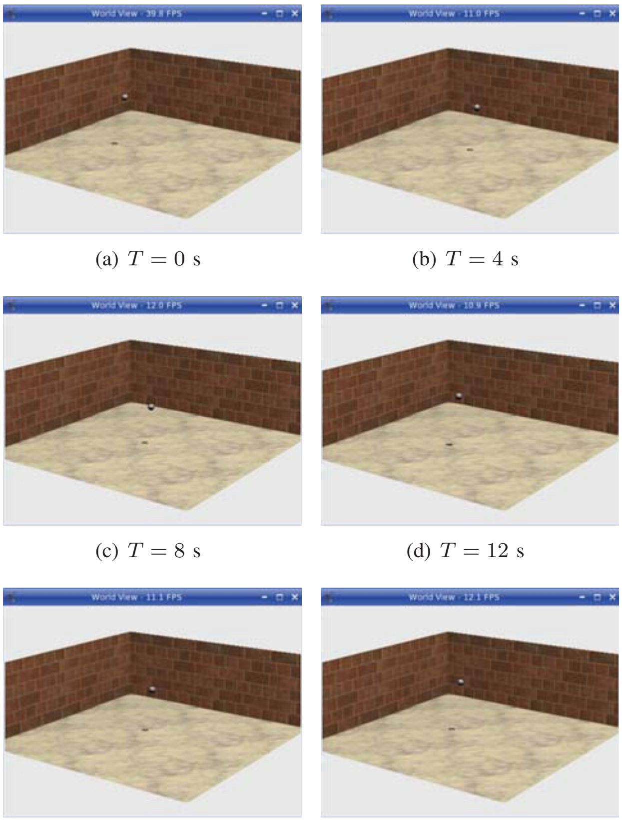
Fig. 7. Reconstructed calibration points with corrected coordinates

Having the the world coordinates correctly set this system also allows the definition of the working field, this increases the safety in the working environment. Any point outside this working area is not updated in the industrial robot.

Fig. 8 shows the virtual world with the detected marker in real-time. This 3D environment is an excellent tool for visually debugging the state of our measured markers [12].

B. Replay movement

The developed software allowed the recording of any shown trajectory. These trajectories could be replayed any



(a) $T = 0$ s (b) $T = 4$ s (c) $T = 8$ s (d) $T = 12$ s (e) $T = 16$ s (f) $T = 20$ s

Fig. 8. Sequence virtualisation of the measured marker

time into the manipulator, hence making this a simple teaching system.

These recorded points can be used in the future to find the best trajectory fitting into lines and curves. This will result in less instructions for the manipulator to execute, thus a faster and smoother movement.

C. Robot communication

At this point any industrial manipulator with a remote control mode can be used with this system. The Motoman controller can receive several types of commands, these can be movement instruction, control commands, memory access, etc.

For this paper the controller basically receives the 3D coordinates of the tool subsequently guiding the manipulator to that same coordinate. Once that position is reached the controller resends a new coordinate s[12]. The coordinates updating rate depends on the speed the manipulator takes to complete its current movement.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

In this paper a simple real-time teaching method for industrial robots by human demonstration was successfully implemented. The system proved to be extremely robust to variations in the environment light making it ideal for real-life applications.

The human operator demonstrated the manipulation of a tool for a stereoscopic vision system, and can see the path mimicked by the robot in real-time. Replaying the extracted points turns this into an immediate teaching machine. The real-time playback is also extremely useful for managing large or dangerous objects.

The use of easily detectable high-intensity and coloured light markers reduced the complexity of using this system by a non-skilled operator.

This system was completely built with standard components which reduces implementation cost, therefore enhancing the reconfigurability of manufacturing systems.

B. Future Works

For this paper a single marker was used to demonstrate the principle, therefore one could only control the tools position. Integrating three or more markers in the tool it will be possible to position and orientate the industrial robot's head.

The development of a path planning method will improve the robot's teaching phase by reducing the number of instructions executed by the robot.

Several precision studies will be done in the future in order to correct more precisely the projective distortions originated from the binocular vision.

A possible application for this technology in virtual reality is also being studied.

VII. ACKNOWLEDGMENTS

Paulo Malheiros acknowledges the Portuguese Science and Technology Foundation (FCT) for his PhD grant.

REFERENCES

[1] Osha Technical Manual, "Section IV: Chapter 4, Industrial Robots and Robot System Safety", http://www.osha.gov/dts/osta/otm/otm_toc.html, 2009.

[2] Robertz, Sven Gestegård and Henriksson, Roger and Nilsson, Klas and Blomdell, Anders and Tarasov, Ivan, "Using real-time Java for industrial robot control", *in JTRES '07: Proceedings of the 5th international workshop on Java technologies for real-time and embedded systems*, 2007.

[3] Yusuke MAEDA, Nanako ISHIDO, Haruka KIKUCHI and Tamio ARAI, "Teaching of Grasp/Graspless Manipulation for Industrial Robots by Human Demonstration", *in Proc. of 2002 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1523-1528, 2002.

[4] J. Norberto Pires, Germano Veiga and Ricardo Araújo, "Programming-by-demonstration in the coworker scenario for SMEs", *in Emerald Industrial Robot*, 2008

[5] The Imaging Source, http://www.theimagingsource.com/, 2009

[6] 5dpo Component Library, http://wiki.lazarus.freepascal.org/5dpo, 2009

[7] Hartley, R. I. and Zisserman, A., *Multiple View Geometry in Computer Vision*, Second Ed., Cambridge University Press, 2004.

[8] Thorsten Thormählen, Hellward Broszio and Ingolf Wassermann, *ROBUST LINE-BASED CALIBRATION OF LENS DISTORTION FROM A SINGLE VIEW*

[9] Hartley, R. I., "In Defense of the Eight-Point Algorithm", *in Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 1997.

[10] Rimon Elias and Robert Laganière, "Projective Geometry for Three-Dimensional Computer Vision", *paper in Viva Research Lab.*

[11] Zhengyou Zhang, "A new multistage approach to motion and structure estimation by gradually enforcing geometric constrains" *in Lecture Notes in Comuter Science 1352*, Computer Vision - ACCV'98, Third Asian Conference on Computer Vi- sion, volume II, pages 567-574, 1998

[12] Real-time teaching videos, http://tinyurl.com/sincrovision, http://www.fe.up.pt/∼robotic/, 2009