Real-time 3D marker tracking with a WIIMOTE stereo vision system: application to robotic throwing

T. Petrič, A. Gams, A. Ude and L. Žlajpah
Department of Automation, Biocybernetics and Robotics
Jožef Stefan Institute
Ljubljana, Slovenia
{tadej.petric, andrej.gams, ales.ude, leon.zlajpah}@ijs.si

***Abstract*—In this paper we describe the use of a standard game console joystick, namely the Nintendo WIIMOTE, for an active real-time 3D marker tracking. We show the ease of applicability of inexpensive and robust standard game controllers for 3D object tracking, e.g. to track an infrared source in 3D space. Recovering the 3D information using stereo vision is still one of the major research areas in computer vision and has given rise to a great deal of literature in the recent past.**

In this paper we present the method for calibrating a WIIMOTE stereo pair without knowing any parameters of the build-in infrared cameras in advance. The results are two matrices which includes both the intrinsic and extrinsic parameters for left and right cameras. The comparison between the stereo and the mono WIIMOTE tracking system is presented. Furthermore, to demonstrate the use of the WIIMOTE stereo system we considered the task of throwing a ball with robotic hand, to the target identified with an infrared source. The throwing task was divided into two separate parts: the tracking part and the throwing part.

***Index Terms*—WIIMOTE stereo, real-time tracking, stereo calibration, robotic throwing.**

I. INTRODUCTION

Recent achievement in the Micro Electronics Mechanical Systems (MEMS) field enabled several developments in today's consumer electronics devices. For example, force plate devices that cost several thousands of euros only few years ago, can now be replaced by inexpensive micro-engineered pressure sensors and can be used in game consoles to train aerobics or exercise yoga. Furthermore, with the rapid development of this technology and decreasing costs, the technology was rapidly adopted in a wide range of devices raging from mobile phones to the computers.

Of particular interest is the appearance of such devices in mass-market products such as games consoles where accelerometers, gyroscopes and different cameras are used to extract user's hand gestures or arm movements [1]. Their low price, wide availability, robustness and several key features that they possess, make them interesting in robotic applications [2].

The two main providers for game controllers are Nintendo with its WIIMOTE controller and Sony with the SIXAXIS controller. When used as an interface, both possess several qualities that make them interesting to be used. Notably, due to their mass product orientation, they are inexpensive, both controllers examined here cost around 50 euros. Moreover, they are widely available and, because they are meant to be used by children, relative solid. Although both controllers show many similarities, only the WIIMOTE is capable of precise movement tracking. By using the embedded one million pixels infrared (IR) camera placed in front the device, the controller can track several IR LED's.

In this paper, we will demonstrate the applicability of two standard WIIMOTE game controllers for real-time 3D marker tracking. We present the method for calibrating a stereo pair, which includes both intrinsic and extrinsic parameters. Furthermore, the calibration parameters are determined without any knowledge of the camera's parameters in advance. The proposed real-time WIIMOTE stereo system can track up to four markers with a sampling rate of 100 Hz.

We show the accuracy of the WIIMOTE stereo system by comparing the obtained 3D position to a known position of a marker in space. For this purpose we have attached a marker to the end-effector of robot and direct kinematic is used to calculate the exact marker position. We also compare these results to the well known WIIMOTE mono tracking system [3] which uses two infrared (IR) sources to calculate the marker position in space. However, the WIIMOTE mono system has some disadvantages.

In order to get an accurate depth of the marker, a strait line between the two IR markers must be parallel to the camera image plane, and it can only track up to two markers. Note also that two IR sources are required to calculate one position of a marker in space.

Next, we show the robotic application of throwing the ball. In this case, the stereo system is attached to the robot and the marker can be anywhere in the field of view. To track the marker the robot has to rotate and the stereo system must simultaneously determine the correct distance to the marker which is necessary to calculate the desired trajectory for throwing the ball to the target.

The paper is organised as follows, in section II we give the description of the stereo system design and the implementation. In section III we present the experimental results for the calibration and the tracking accuracy of the robotic end-effector. In section IV we present the robotic throwing. Conclusion and future work are given in section V.

II. SYSTEM DESIGN AND IMPLEMENTATION

The WIIMOTE's IR camera can detect and track up to four infrared light sources. The camera has a built-in image

978-1-4244-6886-7/10/$26.00 ©2010 IEEE

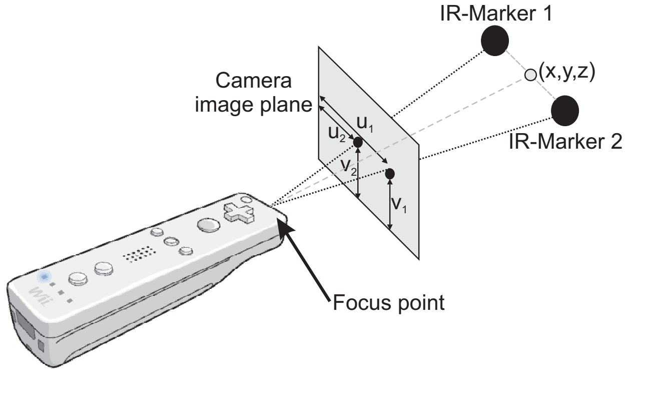

Fig. 1. WIIMOTE mono setup. Projection of an IR markers onto the WIIMOTE camera image plane. Coordinates (u_1, v_1) are the position of a first marker and coordinates (u_2, v_2) are the position of a second marker. Coordinates (x, y, z) represent the tracking point in a global coordinate system. In this figure, the camera's focus point is behind its image plane for clarity. In realty, the focus point is in the center of the camera's lens in the front of the image plane. This does not affect the projection equation.

processor that can analyse the raw camera image, identify bright spots, and compute their positions (u,v) in the camera's image plane. The values can be queried by the host computer through the WIIMOTE's Bluetooth report stream. This is relative straightforward in the sense that its Bluetooth implementation (a BCM2042 chip form Broadcom) complies with the HID standard. The WIIMOTE can be used as mouse emulation or it can be interfaced directly to get the various sensors values, e.g. values form accelerometers, buttons, cameras, etc.

There are at least two possible setups for calculating the exact world position (x,y,z) of a marker. In the first case, only one camera has been used [3]. The marker is represented by two IR beacons with a known distance between them (see Fig. 1. The disadvantage of this setup is that the line between two IR sources has to be parallel to the camera image plane in order to get an accurate result. To overcome this, a setup presented in Fig. 2 has been used. In this setup two IR cameras are used and only one IR source is needed to calculate the exact world position (x,y,z). In Section A we describe the perception problem of how to acquire the world position (x,y,z) of a marker [4], [5], using stereo vision with two WIIMOTEs. In Section B we describe the calibration algorithm of the WIIMOTE stereo system.

A. Acquiring 3D position of an object

As we already mentioned, a 3D position of an object in space is acquired by using two stereo images, based on extrinsic and intrinsic parameters of the cameras.

The intrinsic parameters $\mathbf{M}_{in}$ can be defined as a set of parameters needed to characterise the optic, geometric and digital characteristic of the viewing camera. For a pinhole camera model [6] we need three sets of intrinsic parameters, the perspective projection, the transformation between camera frame coordinates and pixel coordinates and the geometrical distortion introduced by the optics.

The extrinsic parameters $\mathbf{M}_{ex}$ are defined as any set of geometric parameters that identify uniquely the transformation between the unknown camera reference frame and the known reference frame, i.e. the world reference frame. A typical choice for describing the transformation between the camera and the world frame is to use a translation vector $\boldsymbol{T}$ and 3x3 rotation matrix $\mathbf{R}$, which are describing the relative position of origins of two reference frames and the rotation between these two frames.

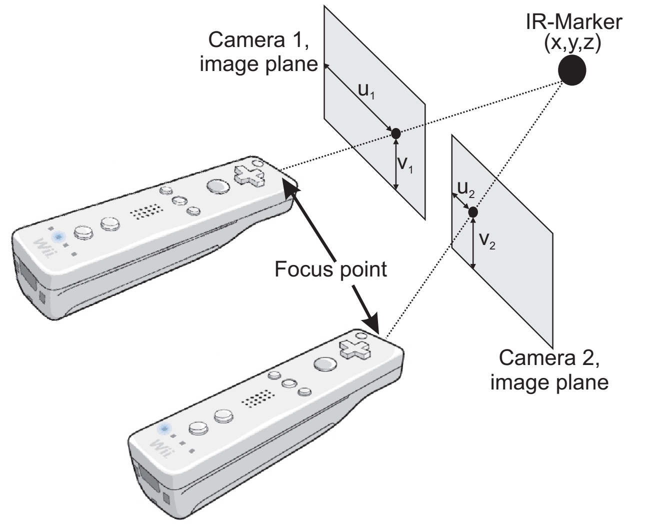

Fig. 2. WIIMOTE stereo setup. Projection of an IR source onto the WIIMOTE cameras' image planes. Coordinates (u_1, v_1) are the position of the marker in camera 1 coordinate system and coordinates (u_2, v_2) are the position of the marker in camera 2 coordinate system. Coordinates (x, y, z) represent the IR source in the global coordinate system. In this figure, the camera's focus point is behind its image plane for clarity. In realty, the focus point is in the center of the camera's lens in front of the image plane. This does not affect the projection equation.

To model the WIIMOTEs cameras we use the standard pinhole camera model. This model is suitable for WIIMOTE camera since it only returns the image plane coordinates (u, v) of an IR source in space. Let denote a 3D point in space by

$$\boldsymbol{p} = \begin{bmatrix} x & y & z \end{bmatrix}^T, \tag{1}$$

and the corresponding 2D points in the left and the right image of the WIIMOTE cameras by

$$\boldsymbol{t_l} = \begin{bmatrix} u_l & v_l \end{bmatrix}^T, \tag{2}$$

$$\boldsymbol{t_r} = \begin{bmatrix} u_r & v_r \end{bmatrix}^T. \tag{3}$$

Furthermore, let $\tilde{\boldsymbol{p}}$, $\tilde{\boldsymbol{t_l}}$ and $\tilde{\boldsymbol{t_r}}$ be the homogenous coordinates of $\boldsymbol{p}$, $\boldsymbol{t_l}$ and $\boldsymbol{t_r}$, respectively. The homogenous coordinates are given by

$$\tilde{\boldsymbol{p}} = \begin{bmatrix} x & y & z & 1 \end{bmatrix}^T, \tag{4}$$

$$\tilde{\boldsymbol{t_l}} = \begin{bmatrix} u_l & v_l & 1 \end{bmatrix}^T, \tag{5}$$

$$\tilde{\boldsymbol{t_r}} = \begin{bmatrix} u_r & v_r & 1 \end{bmatrix}^T. \tag{6}$$

The relationship between a 3D point $\tilde{\boldsymbol{T}}$ and its projection $\tilde{\boldsymbol{t}}_{\boldsymbol{l}}$ in the left image is then given by

$$s_l\tilde{\boldsymbol{t}}_{\boldsymbol{l}} = \mathbf{M}_{in_l}\mathbf{M}_{ex_l}\tilde{\boldsymbol{p}} = \begin{bmatrix} a_{11_l} & a_{12_l} & a_{13_l} & a_{14_l} \\ a_{21_l} & a_{22_l} & a_{23_l} & a_{24_l} \\ a_{31_l} & a_{32_l} & a_{33_l} & a_{34_l} \end{bmatrix}\tilde{\boldsymbol{p}}, \quad (7)$$

and for the right image by

$$s_r\tilde{\boldsymbol{t}}_{\boldsymbol{r}} = \mathbf{M}_{in_r}\mathbf{M}_{ex_r}\tilde{\boldsymbol{p}} = \begin{bmatrix} a_{11_r} & a_{12_r} & a_{13_r} & a_{14_r} \\ a_{21_r} & a_{22_r} & a_{23_r} & a_{24_r} \\ a_{31_r} & a_{32_r} & a_{33_r} & a_{34_r} \end{bmatrix}\tilde{\boldsymbol{p}} \quad (8)$$

Here, s_l and s_r are arbitrary scale factors, $\mathbf{M}_{ex}$ are the extrinsic parameters denoting the rotation and translation that relate the world coordinate system to the camera coordinate system, and $\mathbf{M}_{in}$ is the intrinsic matrix, describing intrinsic camera parameters.

For a standard stereo vision system correspondence is a problem which deals with establishing correspondence between image elements of a stereo pair. The correspondence is a very difficult problem in artificial vision [7]. Since in our experiments we only track one IR source, we will assume that the left and the right points belongs to the same IR source.

Based on above assumptions the position of the marker can be calculated, based on marker positions in both images. Eliminating the scale factors form (7) and (8) yields

$$u = \frac{a_{11}x + a_{12}y + a_{13}z + a_{14}}{a_{31}x + a_{32}y + a_{33}z + a_{34}}, \quad (9)$$

and

$$v = \frac{a_{21}x + a_{22}y + a_{23}z + a_{24}}{a_{31}x + a_{32}y + a_{33}z + a_{34}}. \quad (10)$$

Rewriting (9) and (10) into matrix form and combining left and right camera we get the relation between the positions in the left and right images ($\boldsymbol{t}_l$ and $\boldsymbol{t}_r$) and the 3D position $\boldsymbol{p}$:

$$\mathbf{A}\boldsymbol{p} = \mathbf{B} \quad (11)$$

where $\mathbf{A}$ is given by

$$\mathbf{A} = \left[\begin{array}{ccc} a_{31_l}v_l - a_{11_l} & a_{32_l}v_l - a_{12_l} & a_{33_l}v_l - a_{13_l} \\ a_{31_l}u_l - a_{21_l} & a_{32_l}u_l - a_{22_l} & a_{33_l}u_l - a_{23_l} \\ \hdashline a_{31_r}v_r - a_{11_r} & a_{32_r}v_r - a_{12_r} & a_{33_r}v_r - a_{13_r} \\ a_{31_r}u_r - a_{21_r} & a_{32_r}u_r - a_{22_r} & a_{33_r}u_r - a_{23_r} \end{array}\right]$$

and $\mathbf{B}$ is given by

$$\mathbf{B} = \left[\begin{array}{c} a_{14_l} - a_{34_l}u_l \\ a_{24_l} - a_{34_l}v_l \\ \hdashline a_{14_r} - a_{34_r}u_r \\ a_{24_r} - a_{34_r}v_r \end{array}\right].$$

Using minimal square error (A is not a square matrix) we can solve (11) for $\boldsymbol{p}$

$$\boldsymbol{p} = \mathbf{A}^{\#}\mathbf{B}, \quad (12)$$

where $\mathbf{A}^{\#}$ denotes the left matrix pseudo inverse given by

$$\mathbf{A}^{\#} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T. \quad (13)$$

The error of the 3D position depends on the precision of the object position in both image planes and on the precision of the camera model defined by (7) and (8).

B. Calibration of the camera

When camera is calibrated it is easy to calculate the location of the marker in space. But as we already mentioned, the extrinsic (position and orientation) and intrinsic parameters of the camera are not known in advance, i.e. the camera is uncalibrated. In order to accurately determine the position of the marker in space we have to calibrate both left and right WIIMOTE embedded cameras. Therefore we propose the following algorithm.

The transformation of one point between the world coordinate system and camera coordinate system is defined with (7) and (8). Where the matrix product is determined with

$$\mathbf{M} = \mathbf{M}_{in}\mathbf{M}_{ex} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix}. \quad (14)$$

Here matrix product $\mathbf{M}_{in}\mathbf{M}_{ex}$ is the transformation matrix with elements a_{ij} and represents the calibration of the camera. The matrix includes position and orientation of the camera as well as focal length, pixel size and image center of the camera. In order to identify the elements of $\mathbf{M}$, we rewrite equation (10) into following matrix form

$$0 = \mathbf{C}'\boldsymbol{M}' = \left[\mathbf{C}'_1 \vdots \mathbf{C}'_2\right]\boldsymbol{M}', \quad (15)$$

Where $\boldsymbol{M}'$ is 12 dimensional vector of camera model parameters determined with

$$\boldsymbol{M}' = \left[a_{11}, \ \dots \ a_{14}, \ a_{21}, \ \dots \ a_{24}, \ a_{31}, \ \dots \ a_{34}\right]^T$$

and the matrix $\mathbf{C}'_1$ and $\mathbf{C}'_2$ are given with

$$\mathbf{C}'_1 = \begin{bmatrix} x & y & z & 1 & 0 & 0 & 0 & 0 & -xu & -yu & -zu \\ 0 & 0 & 0 & 0 & x & y & z & 1 & -xv & -yv & -zv \end{bmatrix},$$

$$\mathbf{C}'_2 = \begin{bmatrix} -u \\ -v \end{bmatrix},$$

Equations (15) shows the relation between one point expressed in the world coordinate system and the image coordinate system. Therefore each point in space gives us two equations. Since we are going to normalize the homogeneous coordinates, any scalar multiple of a projective matrix will give the same results [8]. This means that matrix $\mathbf{C}$ has a rank, at most 11 or worse, if the points are badly chosen. It follows that theoretically the minimum number of points in space required to solve (15) is six. However, if n points are used (15) becomes:

$$0 = \mathbf{C}\boldsymbol{M}' = \left[\mathbf{C}_1 \vdots \mathbf{C}_2\right]\boldsymbol{M}', \quad (16)$$

where matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ are given with

$$\mathbf{C}_1 = \left[\begin{array}{ccccccccccc} x_1 & y_1 & z_1 & 1 & 0 & 0 & 0 & 0 & -x_1u_1 & -y_1u_1 & -z_1u_1 \\ 0 & 0 & 0 & 0 & x_1 & y_1 & z_1 & 1 & -x_1v_1 & -y_1v_1 & -z_1v_1 \\ \hdashline \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \hdashline x_n & y_n & z_n & 1 & 0 & 0 & 0 & 0 & -x_nu_n & -y_nu_n & -z_nu_n \\ 0 & 0 & 0 & 0 & x_n & y_n & z_n & 1 & -x_nv_n & -y_nv_n & -z_nv_n \end{array}\right],$$

$$\mathbf{C}_2 = \begin{bmatrix} -u_1 & -v_1 \vdots \ldots \vdots -u_n & -v_n \end{bmatrix}^T.$$

As stated earlier the rank of matrix $\mathbf{C}$ is at most 11. That means that there is no unique solution for the projective matrix $\mathbf{M}$. To solve it within a scaling factor, a_{34} is usually set to 1 and others are calculated by:

$$0 = \begin{bmatrix} \mathbf{C}_1 \vdots \mathbf{C}_2 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{14} \\ a_{21} \\ a_{22} \\ a_{23} \\ a_{24} \\ a_{31} \\ a_{32} \\ a_{33} \\ 1 \end{bmatrix} = \mathbf{C}_1 \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{14} \\ a_{21} \\ a_{22} \\ a_{23} \\ a_{24} \\ a_{31} \\ a_{32} \\ a_{33} \end{bmatrix} + \mathbf{C}_2. \qquad (17)$$

It follows that

$$\begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{14} \\ a_{21} \\ a_{22} \\ a_{23} \\ a_{24} \\ a_{31} \\ a_{32} \\ a_{33} \end{bmatrix} = -\mathbf{C}_1^{\#}\mathbf{C}_2, \qquad (18)$$

where $\mathbf{C}_1^{\#}$ is the left pseudo-inverse of the matrix $\mathbf{C}_1$, determines with (13). However, in the real-world the measurements are noisy, and inaccurate data can make the solutions inaccurate. Therefore, for real application we propose to use at least 30 points (see Fig. 4). Alternative to the pseudo-inverse method is a recursive pseudo-inverse method, where more inaccurate measurements have less influence to the final result. Therefore, better result can be found with the same set of points. However this method is less computationally efficient compared to the pseudo-inverse method.

III. WIIMOTE STEREO CALIBRATION AND TRACKING RESULTS

In this section we discusses the optimal number of points required for calibrating WIIMOTE stereo system, and the differences between the stereo and the mono WIIMOTE tracking systems.

Stereo system was calibrated in a robot's coordinate system as presented in Fig. 3. This can easily be done because in our calibration algorithm the extrinsic parameters are automatically included in the calibration matrix calculation. It means that the position of a stereo system before calibration can be arbitrary as long as the stereo WIIMOTE system detects an IR source. However, after the calibration process has been done, the relationship between the robot's coordinate system and WIIMOTE coordinate system must retain. Otherwise, it is necessary to repeat the calibration process.

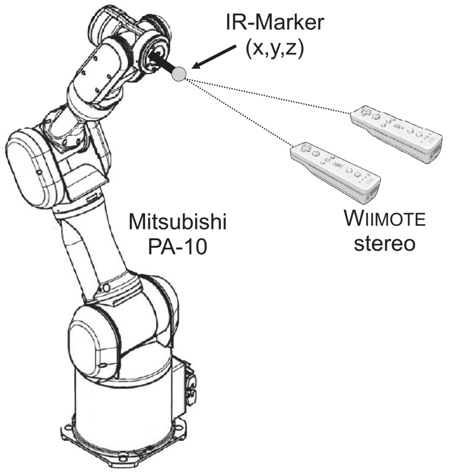

Fig. 3. Experimental setup for WIIMOTE stereo calibration and tracking.

Fig. 4 shows the absolute error as a function of the number of points included in the calibration. The average absolute error is calculated using the "leave one out" cross validation method. Although, theoretically only six points are enough to solve (18), the average absolute error is significantly higher than in the case when more points are used. The error $|e|$ reaches its minimal value when approximately 30 or more points are used. Hence, we proposed that 30 points are enough for calibration of the tracking system.

The calculated values in the robot's coordinate system are presented in Fig. 5. In this case the x axis presents the width from the stereo system's point of view, y axis is the height and on z axis is the depth. As we can see, the calculated and actual values are very well matched, especially in x and y directions, which corresponds to the width and the height direction of the camera. As expected slightly worse is the matching between the actual and the calculated values in the z direction. However, the depth calculation using WIIMOTE stereo system is better than the depth calculation using one

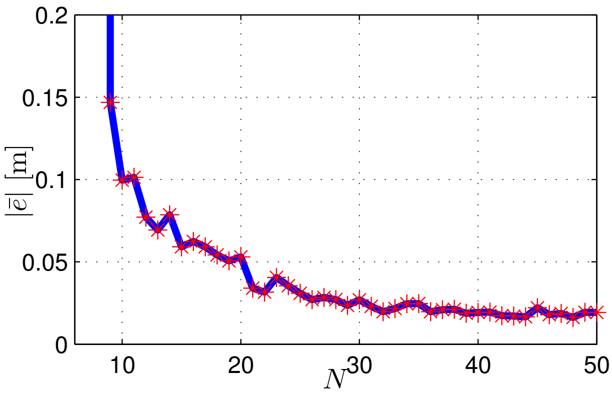

Fig. 4. Comparison between the size of the absolute error as a function of the number of points covered in the calibration.

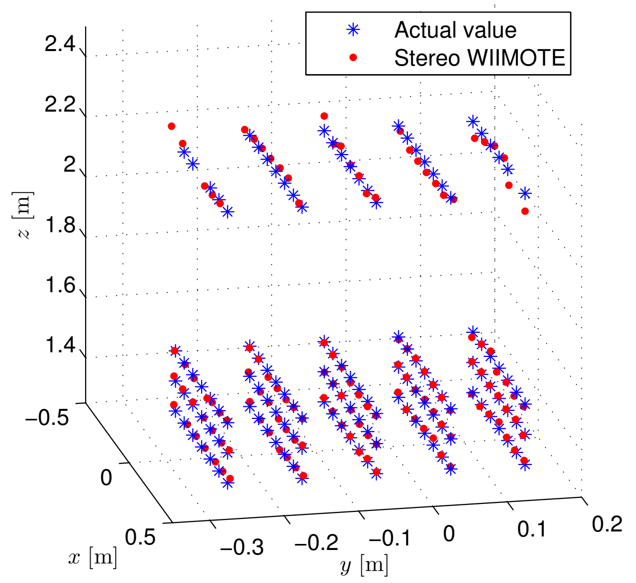

Fig. 5. Data set for WIIMOTE stereo calibration. Actual values are presented with a star marker (∗) and calculated values after calibration are presented with a dot marker.

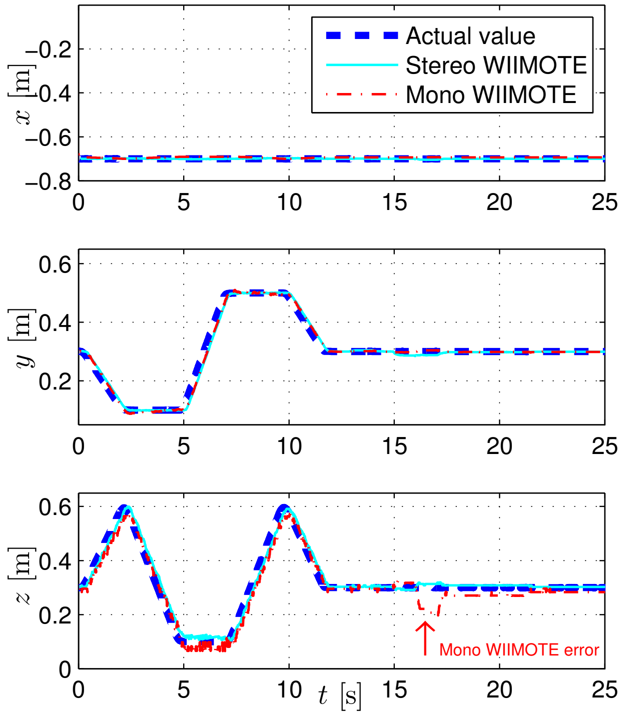

Fig. 6. Comparison between mono and stereo WIIMOTE vision system. Top plot show the comparison in X direction (width), middle plot show the comparison in Y direction (height) and bottom plot show the comparison in Z direction (depth).

WIIMOTE and two IR sources as presented in Fig. 1. The only advantage of the mono system is that only one WIIMOTE is required. The disadvantage of mono WIIMOTE is that two IR markers with a known distance between them are needed, and this system works only when the straight line between two IR sources is parallel to the camera image plane. In any other cases, i.e. the line between markers is not parallel, the calculation of the depth will be inaccurate (as can be observed in Fig. 6).

As in the case of stereo WIIMOTE system the depth calculation is not affected by the rotation of the IR source, therefore we prefer to use for tracking, a WIIMOTE stereo system.

IV. ROBOTIC BALL THROWING TASK

To illustrate the effectiveness of the WIIMOTE stereo system, we selected a task of trowing the ball into a target identified with an IR source. Since the target, i.e. the position where the ball is supposed to land, can be placed at an arbitrary position (x,y,z) in space, we divided the problem into two separate sub-tasks. The first task is the IR source tracking and the second task is hitting the target with the ball.

This reduces the trowing problem form throwing in the space (x,y,z) into the throwing in the plane (x,z), where x is the height coordinate and z is the depth coordinate of the target. This is possible, because the first joint of the robot has been used to turn the robot in a position where y coordinate is zero ($y = 0$). Therefore, the throwing task described in [9] has been used, where only the throwing in the plain was described in a detail.

Our experimental setup consisted of a velocity controlled Mitsubishi PA-10 robot with a mounted Barrett hand and the WIIMOTE stereo system attached to the robot's first joint. The experimental setup is presented in Fig. 7.

To test the throwing accuracy we recorded in the learning phase 25 trajectories, which were manually trained for different targets, and measured where the ball landed with the WIIMOTE stereo system for each of those trajectories. The sequence of tracking and throwing is presented in Fig. 8. Where in the first 10 seconds we can see that the robot is tracking the marker and after that it throws the ball into the target. The repeatability of the throws was approximately 2-3 cm and the accuracy of

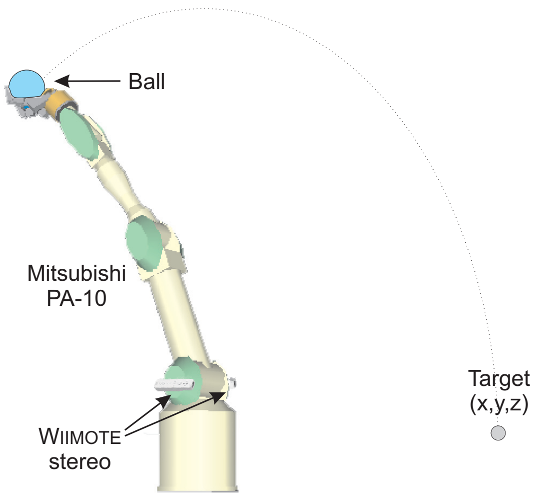

Fig. 7. Experimental setup for robotic ball throwing task.

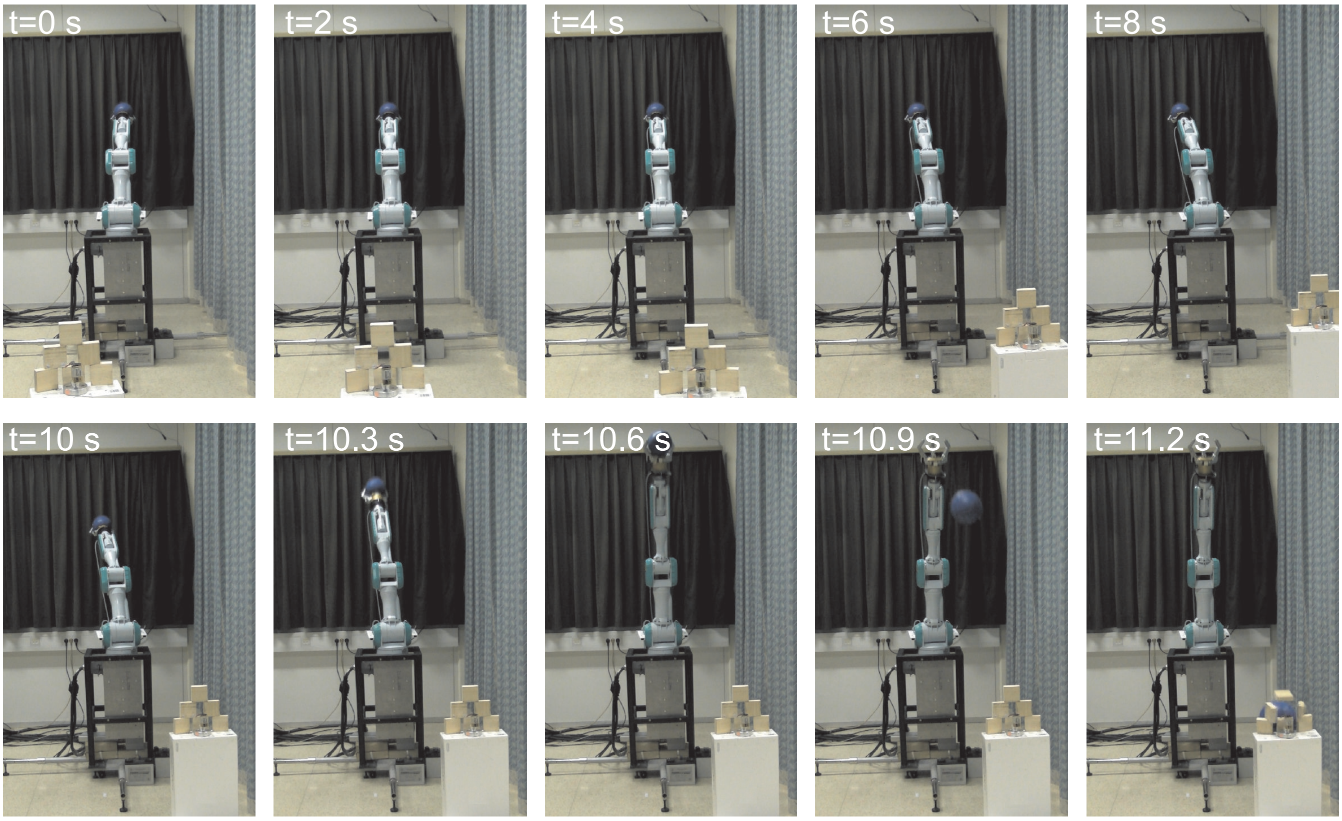

Fig. 8. Image sequence form $t = 0$ s to $t = 10$ s illustrate the tracking process and form $t = 10$ s forward the throwing process.

hitting the desired target marker was about 2-10 cm for the target within the training area.

As the target position measurement error is included in the learning process of the throwing trajectories, the calibration error is compensated by the learning algorithm, despite the error in the absolute marker position in space.

V. Conclusion

In this paper we show how to apply standard consumer electronics components as a 3D tracking system in robotics. We showed a method for calibration of a WIIMOTE stereo vision system with the results of tracking the robotic end effector, marked with an IR source.

Using the Nintendo WIIMOTE, even though it has some drawbacks, is an effective way of tracking markers in space or intuitive control of robots. We expect that off-the-shell electronic devices will prove even more useful in future robotics application. Consequently this will reduce the prices and bring robotic applications closer to everyday use.

Furthermore, the robotic throwing experiment in space was shown as an example of efficiently using a WIIMOTE stereo system in robotics.

References

[1] A. Ude, D. Omrčen, and G. Cheng, “Making object learning and recognition an active process,” *International Journal of Humanoid Robotics*, 2008.

[2] A. Gams and P. Mudry, “Gaming controllers for research robots : controlling a humanoid robot using a wiimote,” in *Zbornik sedemnajste mednarodne Elektrotehnike in računalniške konference ERK 2008, 29. september - 1. oktober 2008, Portorož, Slovenija*. Ljubljana: IEEE Region 8, Slovenska sekcija IEEE, 2008, pp. 191–194.

[3] O. Kreylos, “Oliver kreylos’ research and development homepage - wiimote hacking.” accessed at 18 March, 2010 at http://graphics.cs.ucdavis.edu/~okreylos/resdev/wiimote/index.html.

[4] A. Bernardino and J. Santos-Victor, “Binocular tracking: integrating perception and control,” *Robotics and Automation, IEEE Transactions on*, vol. 15, no. 6, pp. 1080 –1094, dec 1999.

[5] A. Fusiello, E. Trucco, and A. Verri, “A compact algorithm for rectification of stereo pairs,” *Mach. Vision Appl.*, vol. 12, no. 1, pp. 16–22, 2000.

[6] E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998.

[7] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006.

[8] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*. Thomson-Engineering, 2007.

[9] A. Gams, T. Petrič, L. Žlajpah, and A. Ude, “Optimizing parameters of trajectory representation for movement generalization: robotic throwing,” in *Submited to RAAD 2010*, 2010.