Péter Major 1 – Dániel Ayhan 1– dr. Péter Tamás 2
1Student
2Associate professor
1,2 Budapest University of Technology and Economics
Mechanical engineering Faculty

Depth sensors in screening of scoliosis

1. Introduction

The aim of the presented project was developing equipments for screening and early recognizing of scoliosis. Moiré method is usually used in practice, but the correct processing of picture not solved correctly yet. Depth sensors create picture and measure the distance 3D points from the sensors at all pixels. With help of these sensors a 2.5 D model can be created about the human back. There are the information to process for automatic or half automatic diagnosis if scoliosis. The paper presents a solution for this purpose.

2. The philosophy of Kinect

There are four main parts in the Kinect sensor. There is an infra-red projector which projects a pattern of points. (reference[1]). There is a optical copied set of pattern based the patent of PrimeSense named Light Coding (reference[2]). The infra-red interval of wavelength ensures that other effect of light not disturb the situation. There are some problems with sunshine with direct reflection and the surfaces not reflecting enough (left lower corner of figure 1). The other important element is the infra-red camera giving a grayscale picture. Because the different optical axes the distance can be calculated upon the pictures and the position of the IR camera. There is a problem because the geometry there is a shadow on the pictures. There is a zoom optic in the camera to process the different distances. Working distance of camera is about from 0,6m-to 8m.

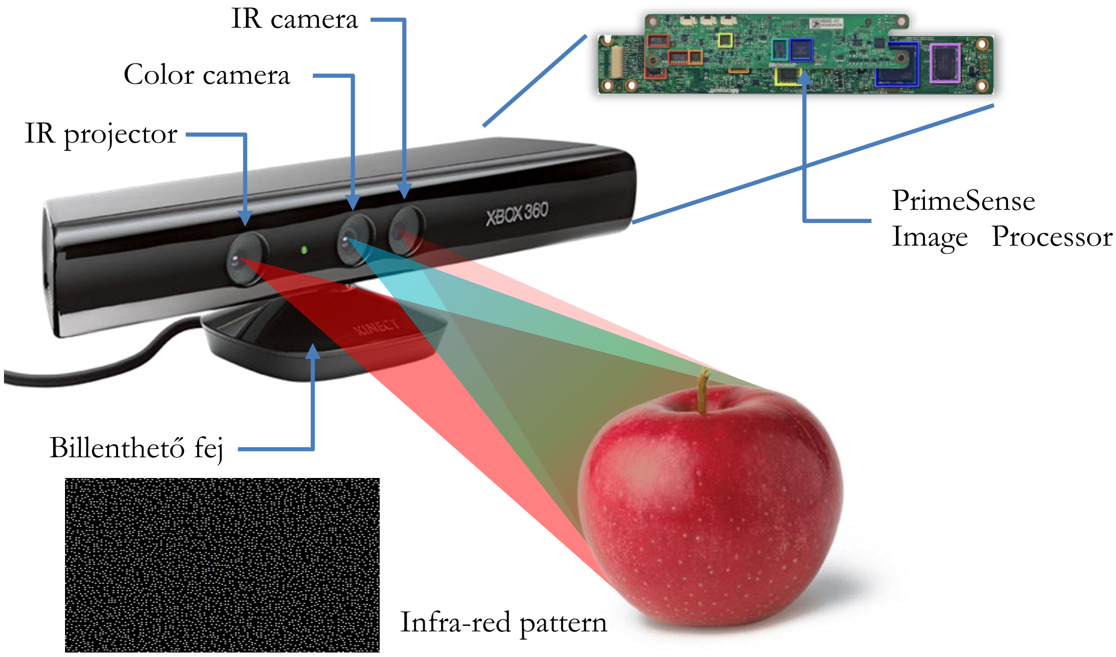


Figure 1.
Elements of Kinect

The third main part of the Kinect is a color camera. Usage the colors ensures creation of textured 3D model. The resolution of cameras is 1280x1024 pixels, because the USB connection the picture processing limited to 640x480 pixels (Figure 2.). In this resolution the communication speed is 30FPS (frame per second). Because the projected net of point is rare the Kinect interpolates the distances. The distances are sent in format of 11 bits.

Figure 2.
The color and IR picture

Microsoft calibrates the cameras in phase of manufacturing, but this calibration not exact enough for 3D position scanning. We prepared our own calibration to solve this problem.

3. Calibration

First we have to define the picture creation of cameras.

3.1 3.1 Projection model of Kinect

The Kinect was modeled as a pinhole camera and the distortion was managed by Brow model. Let a 3D point on the surface of an object is (P)! Let us suppose we know the position of P in word coordinate system ($\boldsymbol{P_w}(x_w, y_w, z_w)$)! The origin of camera at the aperture of the lens, X the horizontal axe, Y the vertical axe of the picture plan and Z is perpendicular to X and Y. In this coordinate system every point has three coordinates (x, y, z). From this we creates z-normed coordinates – where z=1 - ($\boldsymbol{P_n}(x_n, y_n)$). After this the distortion is modeled (reference[2], reference[3]). There is a radial type distortion modeled by (1) and (2) equations.

$$r_n^2 = x_n^2 + y_n^2 \qquad (1)$$

$$\boldsymbol{P_d} = \begin{bmatrix} x_d \\ y_d \end{bmatrix} = (1 + k_1 r_n^2 + k_2 r_n^4 + k_3 r_n^6) \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \boldsymbol{d_t} \qquad (2)$$

Where $\boldsymbol{P_d}(x_d, y_d)$ are the coordinates of point after distortion, $\boldsymbol{P_n}(x_n, y_n)$: are the Z −normed point, k_i coefficient of radial distortions, d_t tangential distortions. Usually only the first two or three used in practice. The tangential distortion can be defined by (3).

$$\boldsymbol{d_t} = \begin{bmatrix} 2t_1x_ny_n + t_2({r_n}^2 + 2{x_n}^2) \\ t_1({r_n}^2 + 2{y_n}^2) + 2t_2x_ny_n \end{bmatrix} \tag{3}$$

Where t_i are the coefficients of tangential distortions. Thes is the Brown model. After the torsion the projection transformation should be defined upon the normalized coordinates. The projection is defined in homogenous coordinate system defined by (4) and (5) equations.

$$\boldsymbol{A} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \boldsymbol{A} \begin{bmatrix} x_d \\ y_d \\ 1 \end{bmatrix} \tag{5}$$

Where $\boldsymbol{A}$ is the camera matrix, f_x and f_y are the focal length in direction x and y, c_x and c_y are the coordinates of the intersection point of the optical axe and the picture plan. Value of s describes the bias of axes from perpendicular case (0). The projected point is $\boldsymbol{P_i}(x_i, y_i)$ in the picture. The picture of projection model is shown on the figure 3. The projection is linear and unambiguous in the direction of arrows in figure 3. In the other direction it is nonlinear and it has more than one solution. Fortunately the GPU (Graphical Processor Unit) will help us to get around this problem.

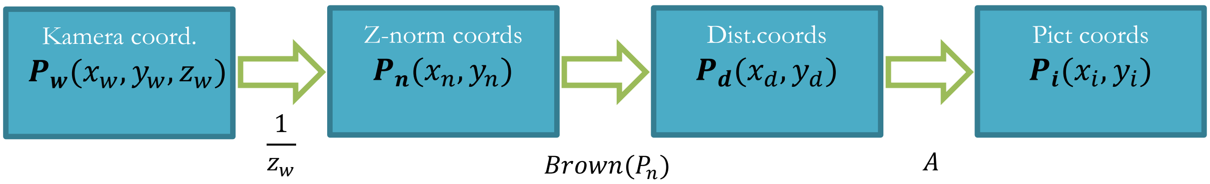


Figure 3.
The projection model

This model was used the color camera and the IR camera also.

3.2 3.2 The calibration process

There are a lot of different methods for calibration (reference[4], reference[5], reference[6]). Becouse the errors of depth picture we started from IR camera. Calibration was done based on Zhang method in Camera Calibration Toolbox in MatLab (reference[7]) by the usual chessboard pattern. The user interface of calibration program is shown on the Figure 4. The prepared application process the pictures created the different positions of chessboard. It defines the corner-points on the tables (see Figure 5.).
Upon defined points the calibration parameters can be computed. The points can be projected back to the space and it is possible to correct some errors. (Figure 6.) The parameters of cameras are defined. To connect them each other the stereo calibration process of Toolbox was used.

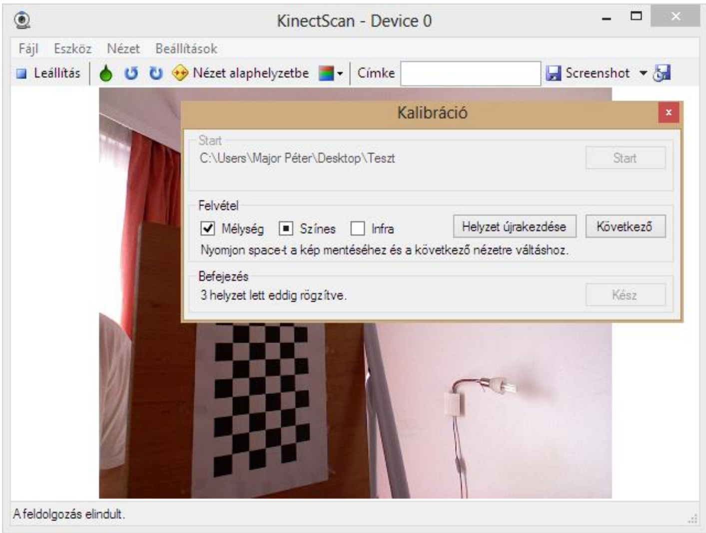
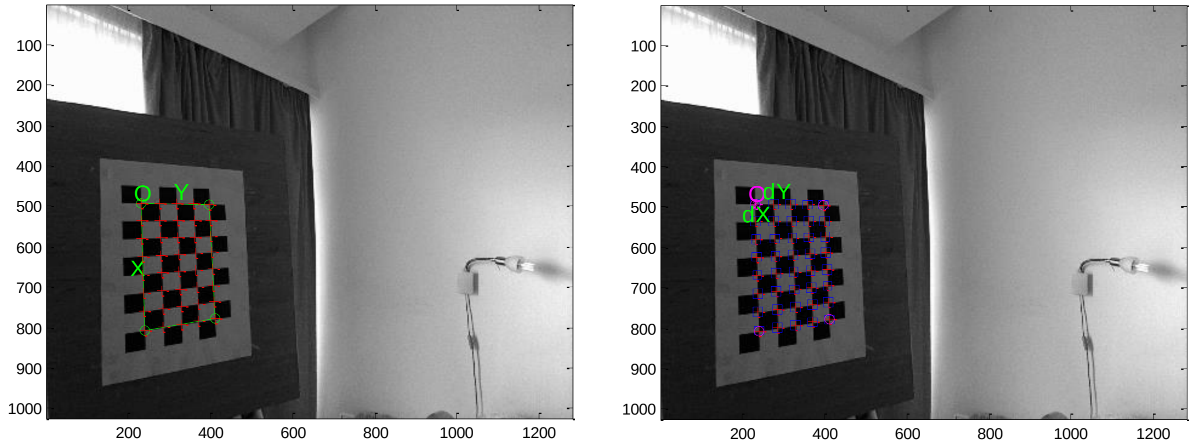
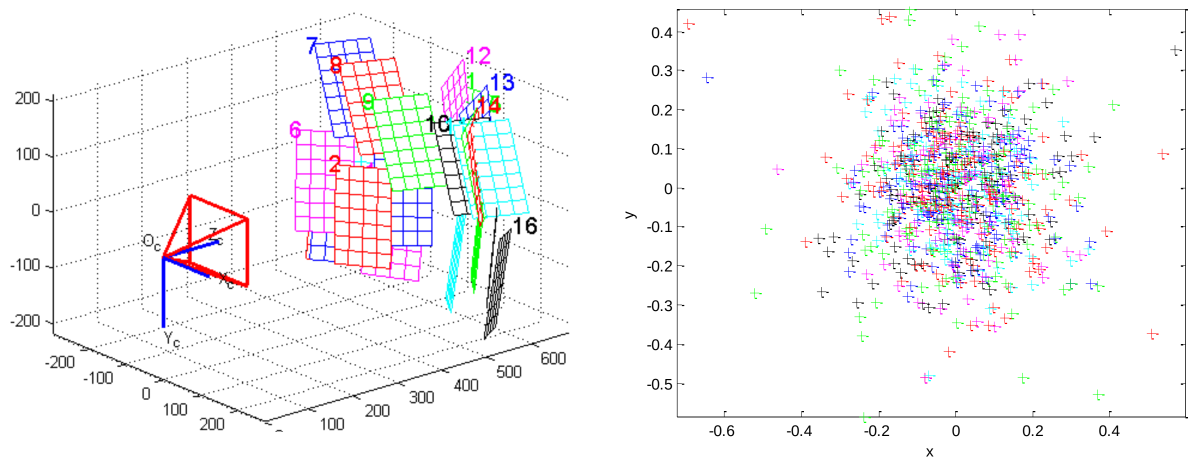


Figure 4.
The calibration process



a) b)

Figure 5.
Defined corners



a) b)

Figure 6.
The projection back

Depth calibration of Kinect was done by a cylinder positioned parallel to the picture plan. Figure 7.shows the measure values. The function can be interpolated by (6).

$$h(x) = -2.33458 - \frac{35498.8}{x - 1093.09} \tag{6}$$

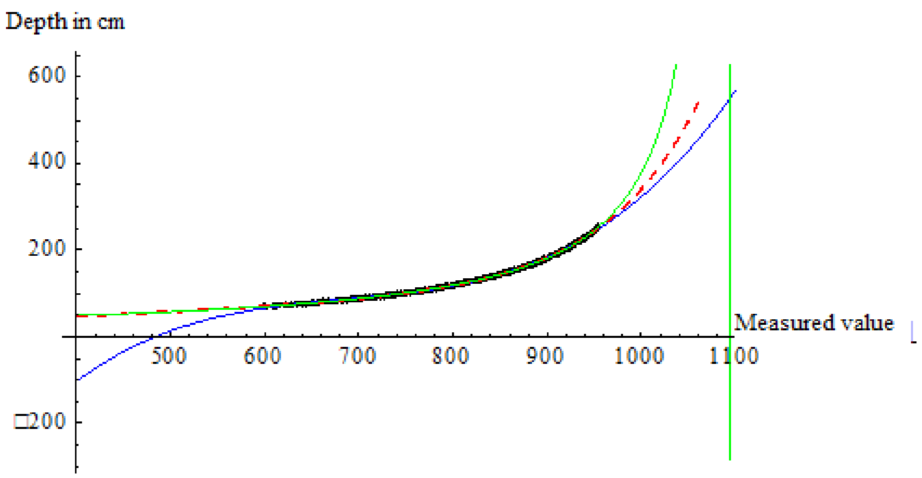


Figure 7.
The depth values

4. Processing of pictures

Result of the presented process is the 3D model of the recorded environment. The process works on real time by help of hardware acceleration.

4.1 Control of process

The driver background of the work is the OpenKinect driver (reference[8]). The working scheme of the driver is shown on Figure 8.

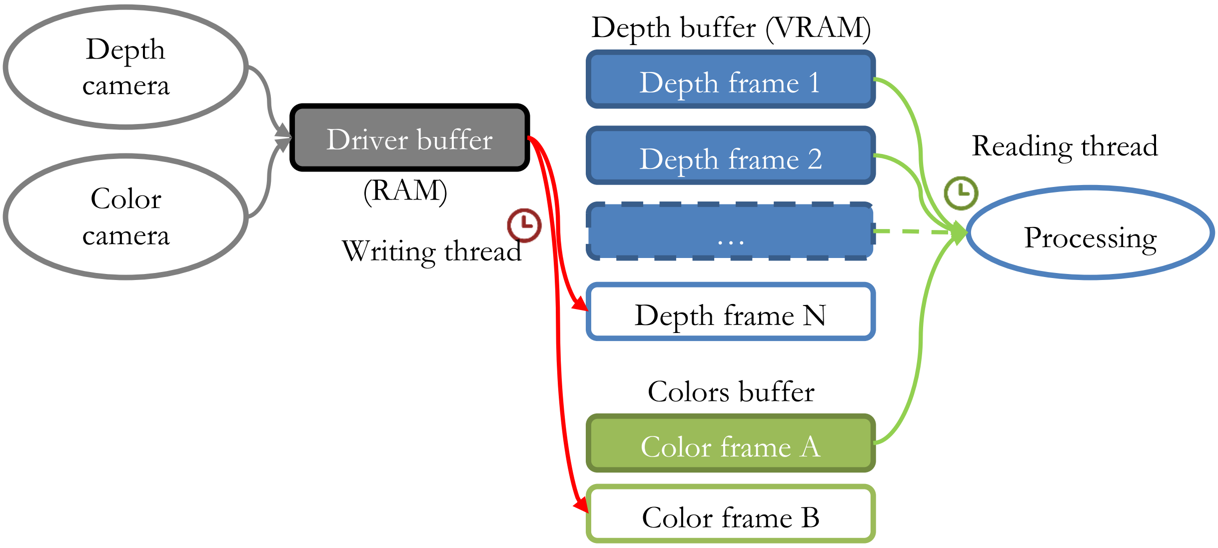


Figure 8.
The working scheme

4.2 Hardware based picture processing

The application development was based on DirectX in .Net environment by system of XNA. Defined vertices of geometry connected by indexes and the texture information were processed in GPU. Data from Kinect are converted to format of GPU. Because the color frames and depth frames arrive not synchronized mode the processing is asynchronous too.
There is a possibility to average the frames arriving one after the other on color picture as well as depth picture.
By calibration defined date of the picture can be corrected in real time. An object in real 3D space can be projected in three steps. The first is the Z normalization, the second is distortion and the last is multiplication by camera matrices. Because of the GPU working philosophy there is a possibility to leave the distortion step.
Because of noise in the pictures there is an optional Gauss filter integrated. It averages the neighbor pixels by the 2D Gauss distribution function.
Creating the 3D model vertex positions defined by the camera model has their own depth value and they can be shown as a 3D surface. The logical scheme of the modeling is shown on Figure 8

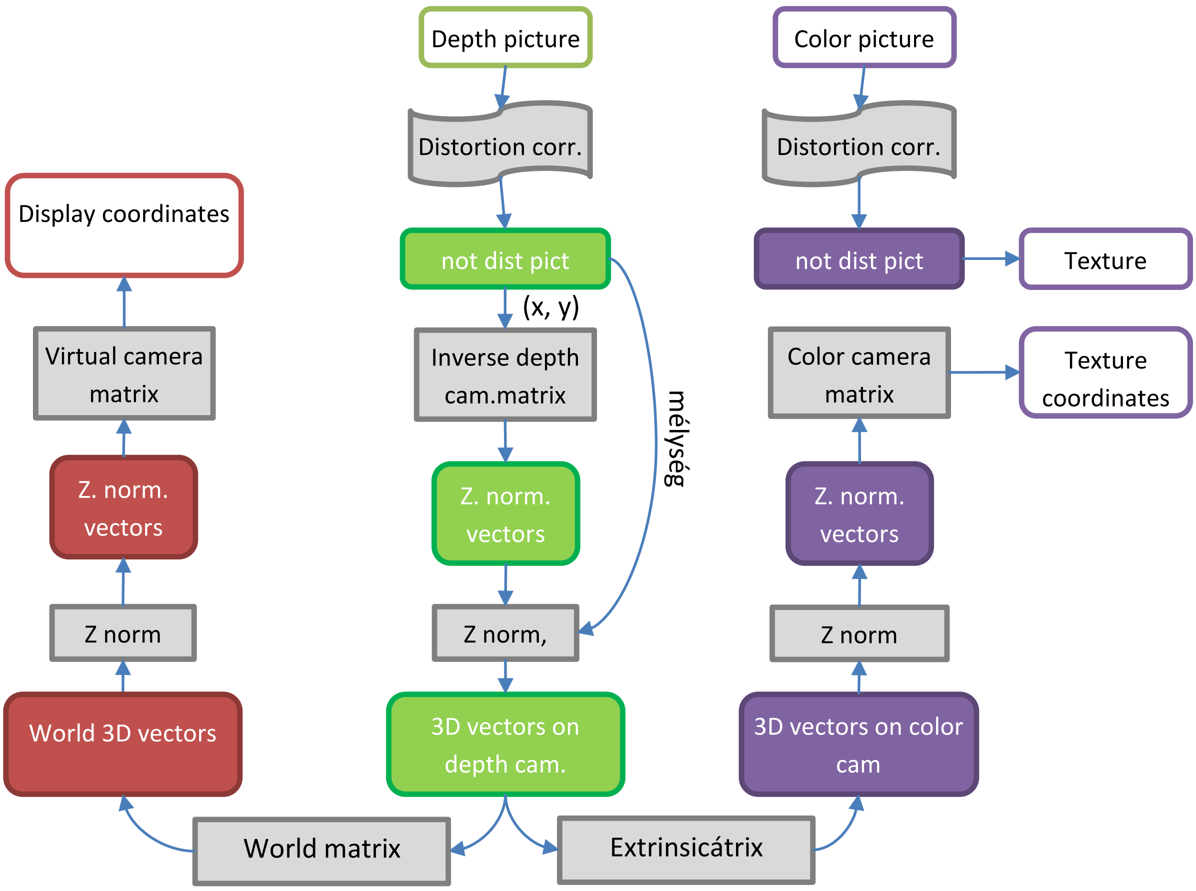


Figure 8.
The working scheme

5. Visualization of model

With the 3D model of object there is a possibility to choose the visualization way. It is possible to put the original texture on the model (first picture in figure 9). Cutting of the model with different depth level and color the surface by black and white in turn can be shown as it was a zebra (second picture in figure 9). We can choose different color for different depth as the picture was a map (third picture in figure 9.) and can use different color between different level in the picture (fourth picture in figure 9).

Figure 9.
The visualization of the model

6. Position of vertebral upon model

There is a hypothesis that the curve of vertebral can be defined upon 3D model. In this case the 3D scanning works as an spinal mouse. In the first step we have to find the symmetry line of the beck. It is computed in some cross-section of the back. The numerical model is based the difference of the two parts as it shown in figure 10. The minimum of the difference as a function of the point position selects the symmetry point in every level.

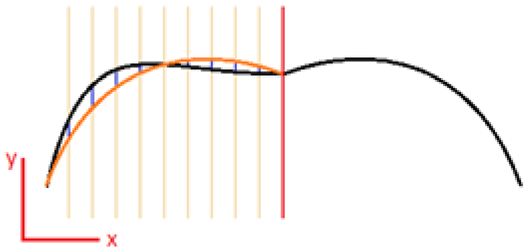


Figure 10.
Searching the symmetry

Upon the hypothesis the symmetry point defines the position of the spine. (Figure 11.)

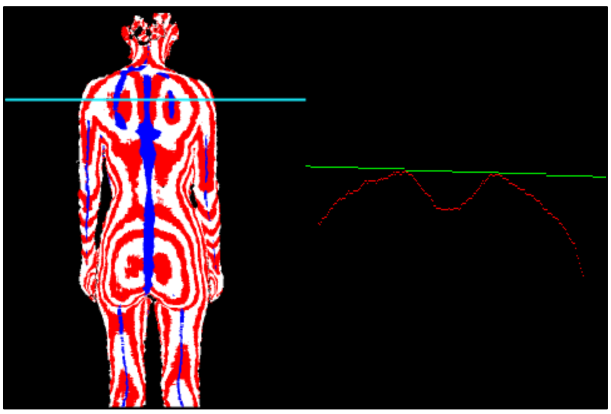


Figure 11.
The spine positions

7. Future work

The presented method can be a good basis of the following tests to prove the possibilities of the method.

8. References

1. Barak Freedman, Alexander Shpunt, Meir Machline és Yoel Arieli: „Depth mapping using projected patterns", United States Patent Application Publication, Pub. No.: US 2010/0118123, 2010
2. D. C. Brown: „Close-Range Camera Calibration", Photogrammetric Engeneering, vol. 37, no. 8, 1971, 855-866. oldal
3. Gergely Vassy and Tamás Perlaki: „Applying and removing lens distortion in post production", The Second Hungarian Conference on Computer Graphics and Geometry 2003, Budapest, 2nd best speech
4. K. Khoshelham: „Accuracy Analysis of Kinect Depth Data", International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XXXVIII-5/W12, 2011, ISPRS Calgary 2011 Workshop, 29-31 August 2011, Calgary, Canada, 133-138. oldal
5. Jan Smisek, Michal Jancosek and Tomas Pajdla: „3D with Kinect", 2011 IEEE International Conference on Computer Vision Workshops, Proceedings, 1154-1160. oldal
6. Daniel Herrera C., Juho Kannala, and Janne Heikkilä: „Accurate and Practical Calibration of a Depth and Color Camera Pair", CAIP 2011, Part II, LNCS 6855, 437–445. oldal
7. Zhengyou Zhang: „A flexible new technique for camera calibration", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 11, 2000, 1330-1334 oldal.
8. „The OpenKinect project", http://openkinect.org/wiki/Main_Page, lekérés időpontja: 2012. 09. 28.