

# Calibration of Kinect-type RGB-D Sensors for Robotic Applications

**Branko Karan**

Institute of Technical Sciences  
Serbian Academy of Sciences and Arts

*The paper presents a calibration model suitable for software-based calibration of Kinect-type RGB-D sensors. Additionally, it describes a two-step calibration procedure assuming a use of only a simple checkerboard pattern. Finally, the paper presents a calibration case study, showing that the calibration may improve sensor accuracy 3 to 5 times, depending on the anticipated use of the sensor. The results obtained in this study using calibration models of different levels of complexity reveal that depth measurement correction is an important component of calibration as it may reduce by 50% the errors in sensor reading.*

**Keywords:** robot vision, Kinect, 3D sensing, depth camera, RGB-D camera, camera calibration.

## 1. INTRODUCTION

Robot system building is a process that regularly involves a number of design compromises. System integrators are often faced with a problem of making a cost-effective system, providing at the same time high accuracy and rich sensing capabilities. Hence, there is an understandable interest in employing consumer-targeted sensing devices with Microsoft Kinect being a notable example. Thanks to its mass production and thus low price, Kinect is perceived as an attractive option in many robotic applications.

Since the initial release of Kinect XBOX 360 in 2010 [1], several variants of the sensor have also appeared. In 2012, Microsoft presented the enhanced Kinect for Windows [2]. At the same time, ASUSTeK and PrimeSense used the same technology to develop 3D vision sensors with the similar features [3, 4] so that it is possible to speak now about a generation of consumer-grade 3D sensors.

Kinect and similar 3D sensors consist of a color (RGB) camera and depth (D) sensor. Generally, they are low accuracy and low precision devices. Diverse studies, including this one, show that their accuracy is on the order of 2-3%. At a distance of 4m from the sensor, this corresponds to RMS error on the order of 10cm. This level of accuracy is quite satisfactory in e.g. human interaction applications. However, it may appear unsuitable in some specific robotic uses (e.g., indoor navigation or fine manipulation).

Accuracy can be improved by software correction of sensor outputs. The correction is based on a specific calibration model whose parameters are identified during the calibration process. The calibration procedure consists in essence in collecting sensor outputs and comparing them to reference data, and it assumes using a special calibration rig, i.e. an object of precisely known dimensions, and/or use of a high-

precision measurement tool.

This paper contributes to software-based techniques for enhancing Kinect sensor accuracy in three ways. First, it presents a calibration model suitable for software calibration of Kinect-type sensors. Second, it describes a two-step calibration procedure recently proposed by this author [5, 6] for identification of the model parameters. The procedure is simple in the sense that it makes use of only a commonly adopted camera calibration tool (i.e., a checkerboard) and it does not require additional specific calibration objects or external measurement devices. Finally, the paper contains a calibration example involving two use cases: one in which only the depth sensor is employed, and another in which a joint use of both RGB camera and depth sensor is assumed. The case study incorporates calibration and accuracy analysis of Kinect XBOX 360 sensor. However, the results and the approach might be quite applicable to other similar sensors.

The paper is written as an extended version of the conference paper [5]. Compared to [5], this paper presents a derivation of the Kinect depth measurement correction model and it additionally provides few more details about the calibration process.

The paper is organized as follows. A short review of several representative works on accuracy and calibration of Kinect-type sensors is given in the next section. Section 3 contains a description of employed sensor calibration model and model identification procedure. Section 4 presents analysis of accuracy that can be expected in different scenarios and with different levels of complexity of calibration models. Section 5 summarizes conclusions on the attained results.

## 2. RELATED WORK

Accuracy of Kinect-type sensors was a subject of investigation involving use of different measuring devices. Dutta [7] measured accuracy of Kinect sensor using a high-precision seven-camera Vicon 3D motion capture system and reported mean errors of up to 10cm (standard deviation on the order of 5cm) in the range covering the distances of up to 3m from the sensor. The sensor was previously calibrated and the calibration

Received: September 2014, Accepted: December 2014

Correspondence to: Branko Karan  
Institute of Technical Sciences SASA,  
Knez Mihailova 35, 11000 Belgrade, Serbia  
E-mail: branko.karan@itn.sanu.ac.rs

doi:10.5937/fmet1501047K

© Faculty of Mechanical Engineering, Belgrade. All rights reserved

FME Transactions (2015) 47, 47-54

47

involved identification of parameters of the sensor camera. Gonzales-Jorge et al. [8] investigated accuracy of uncalibrated Kinect XBOX 360 and Asus Xtion using a specially designed measurement fixture. Their measurements confirm that both accuracy and precision deteriorate with the distance. These authors obtained similar accuracy/precision for both examined sensors and the values of RMS error were on the order of 10mm (standard deviation on the order of 8mm) at the distance of 2m.

Several works addressed improvement of sensor accuracy. In earlier works, e.g. Burrus [9] and Zhang and Zhang [10], the main concern was on procedures and methods for identification of intrinsic parameters of sensor cameras. Later, the focus has moved to calibration of sensor depth measurement model, with important works of Khoshelham and Elberink [11], Smisek et al. [12], and Herrera C. et al. [13]. These works addressed transformation of disparity maps provided by the sensor into depth maps. However, with the actual OpenNI [14] and Microsoft Kinect SDK [15] application programming interfaces, disparity data are already converted into depth using a nominal factory model. To account for this change, a reformulation of depth calibration model was proposed in [6], advocating to utilize a linear relationship between actual and sensor-provided inverse depths.

Calibration of Kinect-type 3D sensors is naturally split into two parts: identification of parameters of sensor cameras and identification of parameters of depth measurement model. It is natural and simplest to identify camera parameters from raw camera data, but there were also attempts to calibrate sensor camera directly from depth maps [10] and perform a joint depth/RGB camera calibration [13]. Although advantageous in principle, this approach suffers from the problem of calibrating the camera using a low precision depth map: the low precision problem leads to a need for an extremely large number of measurements. Additionally, the joint calibration, although having a potential of improving the optimal solution, may display an undesired interaction: for example, the data on joint calibration reported in [13] show that a refinement of depth model can paradoxically lead to enlargement of reprojection errors of RGB camera.

An important issue in depth model calibration is the measurement of depth as it normally requires either a special 3D calibration rig or an external measuring tool. In [11], depth was measured using a simple measuring tape. In [13], correspondence between depth map and RGB camera image was established using external corners of calibration table. A similar approach was proposed by Draelos et al. [16]. Geiger et al. [17] proposed more complex calibration object, consisting of multiple checkerboards. Shibo and Qing [18] designed a specific calibration board with regularly-spaced drilled holes allowing their easy identification in both RGB images and depth maps.

In this work, depth calibration follows a procedure proposed in [6], where the Kinect RGB camera, once calibrated, is used as a depth measuring device for subsequent calibration of the depth model. Details of

both the model and the calibration procedure are given in the next section.

### 3. SENSOR CALIBRATION

Kinect-type 3D sensors considered in this work operate as structured light sensors. A sensor (Fig. 1) incorporates a laser infra-red (IR) diode for emitting a dotted light pattern and an IR camera for capturing reflected patterns. Depth is calculated by sensor software on the basis of disparity of reflected patterns with the respect to the reference patterns obtained for a plane placed at a known distance from the sensor. A supplementary RGB camera is added to provide additional information on color and texture of the surface. Thus, sensor output consists of three data flows: images from RGB camera, raw images from IR camera, and depth maps calculated by sensor firmware. Sensor calibration can be viewed as a refinement of correspondences between 3D object coordinates and coordinates in RGB, IR, and depth images.

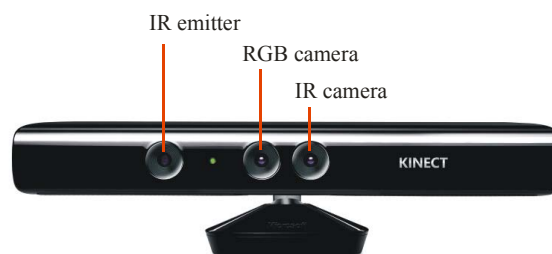


Figure 1. Kinect sensor

The proposed calibration procedure consists of two steps: the first step comprises calibration of sensor's RGB/IR cameras, whereas the calibration of depth model is performed within the second step.

#### 3.1. Camera calibration

Camera calibration assumes identification of parameters of functions modeling transformation of 3D coordinates of external objects into coordinates in image plane. Given 3D homogeneous coordinates  $\mathbf{X}_e = [X_e, Y_e, Z_e, 1]^T$  of a point in some external world coordinate frame, the transformation of coordinates involves:

- The transformation  $\mathbf{X} = \mathbf{T}_e \mathbf{X}_e$  into coordinates  $\mathbf{X} = [X, Y, Z, 1]^T$  in camera frame,
- The projection  $Z \cdot \mathbf{x} = [\mathbf{I}, \mathbf{0}] \mathbf{X}$  into the point  $\mathbf{x} = [u, v, 1]^T$  in normalized image plane,
- The distortion  $\mathbf{x} = f^{(d)}(\mathbf{x})$ , yielding distorted normalized coordinates  $\mathbf{x}^{(d)} = [u^{(d)}, v^{(d)}, 1]^T$ , and finally,
- The transformation into pixel coordinates, using transformation of the form  $\mathbf{x}^{(p)} = \mathbf{K} \cdot \mathbf{x}^{(d)}$ , where  $\mathbf{x}^{(p)} = [c, r]^T$  and  $c, r$  are column/row indices of image pixels.

Thus, calibration consists in identification of:

- *Intrinsic parameters:* elements of the camera matrix  $\mathbf{K}$  and distortion function  $f^{(d)}(\cdot)$ , and
- *Extrinsic parameters:* elements of the matrix  $\mathbf{T}_e$ .

For a stereo pair of Kinect cameras, calibration encompasses identification of camera matrices  $\mathbf{K}_{RGB}$ ,  $\mathbf{K}_{IR}$ , distortion functions  $f_{RGB}^{(d)}(\cdot)$ ,  $f_{IR}^{(d)}(\cdot)$ , and homogeneous transformation matrix  ${}^{IR}\mathbf{T}_{RGB}$  for transformation of homogeneous 3D coordinates from the coordinate frame of RGB camera to the coordinate frame of IR camera.

In this work, the following functional forms of distortion function and camera matrix were considered (see e.g. [19] for details):

$$f^{(d)}(\mathbf{x}) = (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \cdot \mathbf{x}, \quad r^2 = \sqrt{u^2 + v^2} \quad (1)$$

$$\mathbf{K} = \begin{bmatrix} f_x & \alpha \cdot f_x & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

It is readily seen that the model (1) neglects tangential distortions. This simplification is made since the early tests with Kinect revealed that tangential distortion of its RGB camera was below the achievable level of precision. Besides, the distortion found in IR camera was extremely low and neglecting the tangential distortion did not change much the total reprojection error. Thus, the adopted functional form (1) involved only a radial distortion specified by the parameters  $k_1, k_2, k_3$ .



a. RGB images



b. IR images

**Figure 2. Close-up views employed in camera calibration**

Identification of RGB/IR camera parameters was conducted in Matlab environment, using Bouguet's camera calibration toolbox [19]. A  $9 \times 8$  checkerboard with 30mm square fields was employed as a calibration rig. Then, a set of ten pairs of close-up RGB/IR images of the checkerboard placed in different positions and orientations were collected and submitted to calibration (Fig. 2). To achieve appropriate light conditions for calibrating the IR camera, the IR emitter was disabled during imaging (an ordinary stick tape was used to cover the projector; the newer Kinect for Windows sensor model allows programmable control over the IR emitter).

While acquiring images, both cameras were set to their maximum resolutions, which were  $1280 \times 960$  for the RGB camera and  $640 \times 480$  for the IR camera.

The results are summarized in Table 1. The most important difference between nominal and identified parameters is in focal length which differs about 1.8% for the RGB camera and 2.4% for the IR camera.

**Table 1. Camera parameters**

	RGB camera		IR camera	
	Nominal	Identified	Nominal	Identified
$f_x$	1062.3	1043.2	571.26	585.5
$f_y$	1062.3	1044.0	571.26	586.5
$c_x$	639.5	650.8	319.5	327.9
$c_y$	479.5	510.4	239.5	246.2
$\alpha$		0.00122		0.00130
$n_x$	1280		640	
$n_y$	960		480	
$k_1$		0.224		-0.125
$k_2$		-0.715		0.438
$k_3$		0.751		-0.556
Rotation vect. RGB→IR: [ 0.00260, -0.00600, 0.00175 ]				
Translat. vect. RGB→IR: [-25.07450, 0.28102, 0.79722 ]				

Once the parameters of transformations are obtained, depth map images provided by the sensor are easily converted into 3D maps. Assuming that a point in the depth map is available in of the form  $(\mathbf{x}_{IR}^{(p)}, z_{IR})$ , where  $\mathbf{x}_{IR}^{(p)}$  are its (column, row) coordinates in the IR image coordinate frame and  $z_{IR}$  is the distance from the sensor, the transformation is given by:

$$\mathbf{x}_{IR} = \left[ f_{IR}^{(d)} \right]^{-1} \left( \mathbf{K}_{IR}^{-1} \cdot \mathbf{x}_{IR}^{(p)} \right) \quad (3)$$

$$\mathbf{X}_{IR} = \begin{bmatrix} z_{IR} \mathbf{x}_{IR} \\ 1 \end{bmatrix} \quad (4)$$

Besides, the depth map can be extended with texture and color information from RGB image using the coordinate transformations:

$$\mathbf{X}_{RGB} = {}^{IR}\mathbf{T}_{RGB}^{-1} \cdot \mathbf{X}_{IR} \quad (5)$$

$$\mathbf{x}_{RGB}^{(p)} = \mathbf{K}_{RGB} \cdot f_{RGB}^{(d)} \left( \frac{1}{Z_{RGB}} [\mathbf{I} \quad \mathbf{0}] \mathbf{X}_{RGB} \right) \quad (6)$$

where  $Z_{RGB}$  is the  $z$ -coordinate of  $\mathbf{X}_{RGB}$ .

Evaluation of expressions (3-6) involves computation of distortions and therefore it is of interest to explore whether a simpler approximation of distortions is possible and whether the distortions could be perhaps completely neglected. Additional insight into the actual level of distortions introduced by camera optics provides Fig. 3, where the amount of distortions, expressed in pixels, is shown on contour lines and the direction of distortions is shown by blue arrows. It is

seen that the direction of distortions is opposite for the RGB and IR cameras, so that they effectively affect the net deviation in the same direction. Distortions increase from the center to periphery of images: if the object of interest is kept within the central circles on Fig. 3, the distortions introduced by cameras could be as low as 1 pixel for the RGB camera and 0.25 pixels for the IR camera. Since the distortions are multiplied to 3D deviations with the factor of  $z/f$ , these pixel distortions correspond to 3D deviations on the order of, respectively, 3.8/1.7mm for the RGB/IR camera at the distance of 4m from the sensor. Thus, the deviations could be neglected for such central objects. On the other hand, by approaching peripheral image area, the deviations enlarge and a more complex model of deviations becomes necessary.

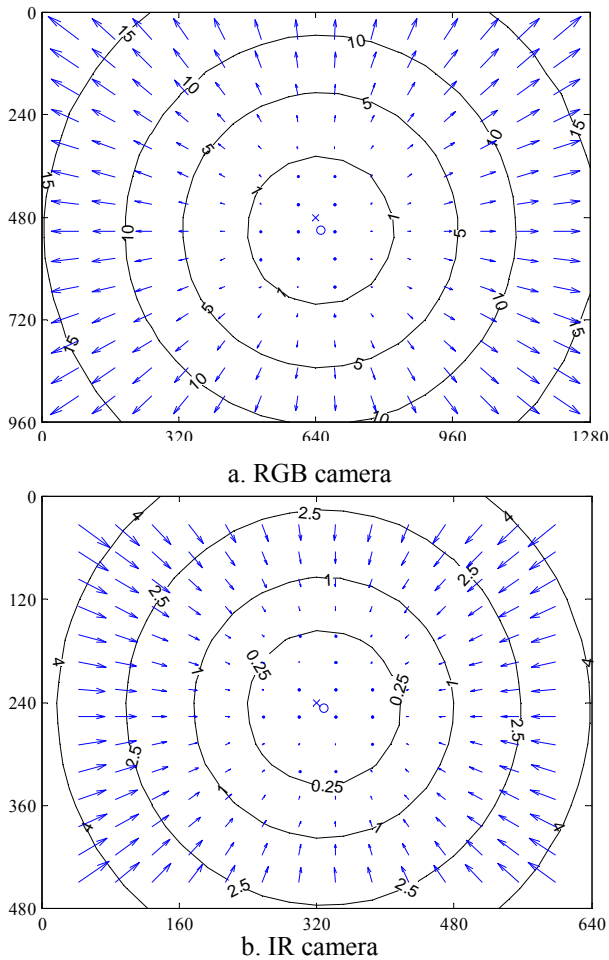


Figure 3. Radial distortions

### 3.2. Depth measurement calibration

Sensor reading of Kinect-type sensors is based on internal computation of depth using the detected disparity between images of IR light beams obtained after reflection from the measurement surface and the reference surface. The value of such inferred depth depends on sensor geometry and optical characteristics that are subject to manufacturing variations.

To derive a relationship between the disparity and the measured depth, consider Fig. 4 that schematically displays a beam  $PMC$  emitted from the projector  $P$

under the angle  $\alpha$ , reflected from the measurement surface at the point  $M$ , and hit the image plane at the point  $C$ , forming the angle  $\beta$  with the sensor optical axis. Looking at Fig. 4 we obtain:

$$l = z \tan \alpha + z \tan \beta \quad (7)$$

$$\tan \beta = \frac{x}{f} \quad (8)$$

where  $l$  is the length of the baseline between the projector and the IR camera,  $z$  is the distance from the reflection surface,  $x$  is the offset of the reflected beam on the sensor surface, and  $f$  is the focal length of the IR camera. By combining (7) and (8), after rearrangement we obtain:

$$\frac{1}{z} = \frac{\tan \alpha}{l} + \frac{x}{lf} \quad (9)$$

For a reference surface at the distance of  $z_0$ , the relation (9) becomes:

$$\frac{1}{z_0} = \frac{\tan \alpha}{l} + \frac{x_0}{lf} \quad (10)$$

By combining (9) and (10), the measurement model is obtained in the form:

$$\frac{1}{z} = \frac{1}{z_0} + \frac{d}{lf} \quad (11)$$

where  $d = x - x_0$  is the disparity between offsets obtained for a measurement and reference surfaces.

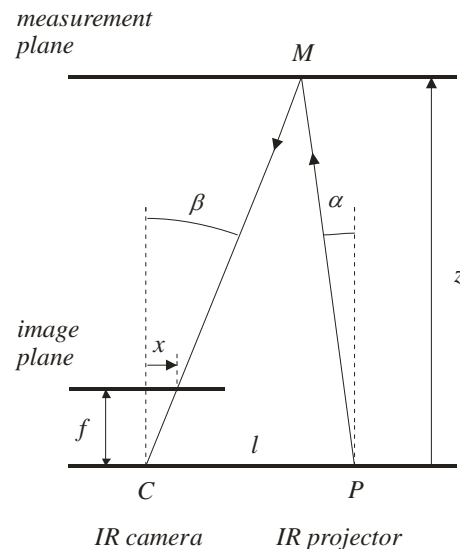


Figure 4. Depth measurement geometry

The model (11) is internally used by sensor software to transform detected disparity into depth. It is readily seen that the inferred value depends on  $l, f, z_0$  that are subject to manufacturing variations. Thus, the actual output  $z_s$  from the sensor is really an approximation based on the nominal model:

$$\frac{1}{z_s} = \frac{1}{z_{0N}} + \frac{d}{l_N f_N} \quad (12)$$



From (11) and (12), the following relation between  $z$  and  $z_S$  is obtained:

$$\frac{1}{z} = a_Z \cdot \frac{1}{z_S} + b_Z \quad (13)$$

where  $a_Z, b_Z$  are the values that are characteristics of particular sensor:

$$a_Z = \frac{l_N f_N}{l f}; \quad b_Z = \frac{1}{z_0} - a_Z \frac{1}{z_{0N}}$$

Ideally,  $a_Z = 1$  and  $b_Z = 0$ . However, for a particular sensor, depth model parameters  $a_Z, b_Z$  may differ from ideal values and it leads to systematic errors in depth measurement. Therefore, appropriate tuning of depth model parameters may improve the accuracy.

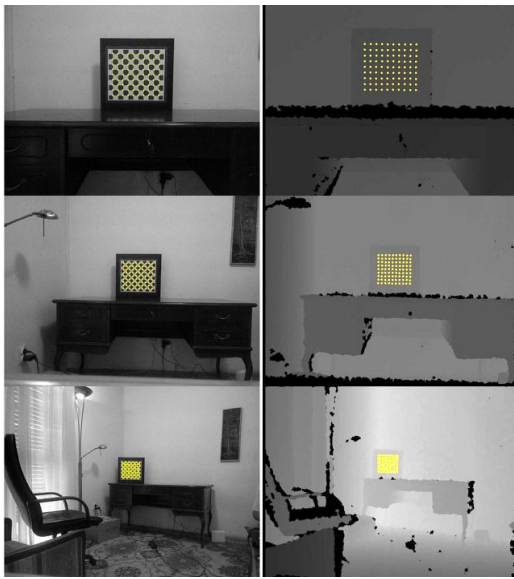


Figure 5. Views employed in depth calibration

In this work, identification of parameters  $a_Z, b_Z$  is conducted by the procedure proposed in [6]. First, a set of pairs of RGB/depth images of the same checkerboard that was used in camera calibration is collected. However, compared to the camera calibration step where a set of close-up views were employed, here a set of views with different object distances are selected. Note yet another difference: in the camera calibration case, the pairs of RGB/IR images were acquired whereas for the depth calibration the pairs of RGB/depth images are collected. Each of RGB images is afterward converted into grayscale and then the corner coordinates are extracted for inner  $10 \times 9$  corners. See Fig. 5 as an example where the extracted corners are emphasized in both RGB and depth images. For each view, say  $k^{\text{th}}$ , the extracted pixel coordinates  $\mathbf{x}_{RGB}^{(p)}(i, j, k)$  of a corner in  $i^{\text{th}}$  row and  $j^{\text{th}}$  column are, using the identified model of the RGB camera, transformed into external coordinates  $\mathbf{X}_{RGB}^{(e)}(i, j, k)$  which are, together with known corner coordinates  $\mathbf{X}_C(i, j) = [w(i-1), h(j-1), 0, 1]^T$  in local checkerboard coordinate frame, where  $w, h$  denote the width/height of the checkerboard fields, used to infer the

position and orientation of the checkerboard  ${}^{RGB}\mathbf{T}_C(k)$  with respect to the RGB camera. Using the known transformation between RGB and IR cameras, corner coordinates are further transformed to IR camera frame as:

$$\mathbf{X}_{IR}(i, j, k) = {}^{IR}\mathbf{T}_{RGB} \cdot {}^{RGB}\mathbf{T}_C(k) \cdot \mathbf{X}_C(i, j) \quad (14)$$

The  $z$ -component  $Z(i, j, k)$  of  $\mathbf{X}_{IR}(i, j, k)$  is afterward compared to sensor reading. The sensor value is determined by converting  $\mathbf{X}_{IR}(i, j, k)$  into pixel coordinates of IR camera and by searching for the nearest neighbor in sensor depth map  $m_k$ :

$$\mathbf{x}_{IR}^{(p)}(i, j, k) = \mathbf{K}_{IR} \cdot f_{IR}^{(d)}(\mathbf{x}_{IR}(i, j, k)) \quad (15)$$

$$Z_S(i, j, k) = m_k(\text{round}(\mathbf{x}_{IR}^{(p)}(i, j, k))) \quad (16)$$

Finally, the obtained set of pairs  $Z(i, j, k), Z_S(i, j, k)$  is employed to estimate the parameters  $a_Z, b_Z$  of depth measurement model (13) using the least squares fit.

Calibration was performed using three views displayed in Fig. 5 and the resulting values of parameters were  $a_Z = 0.9969$  and  $b_Z = 4.2881 \cdot 10^{-6}$ .

Fig. 6 illustrates the depth correction curve obtained for these values (the curve shown in black). It is seen that the correction increases with depth up to the value on the order of 55mm at the end of the sensor range. For reference, Fig. 6 also shows measurement errors obtained for all checkerboard corners in all views in Fig. 7. The measurement points are shown as green (lighter) dots; the points that were used in calibration are highlighted in red (darker dots).

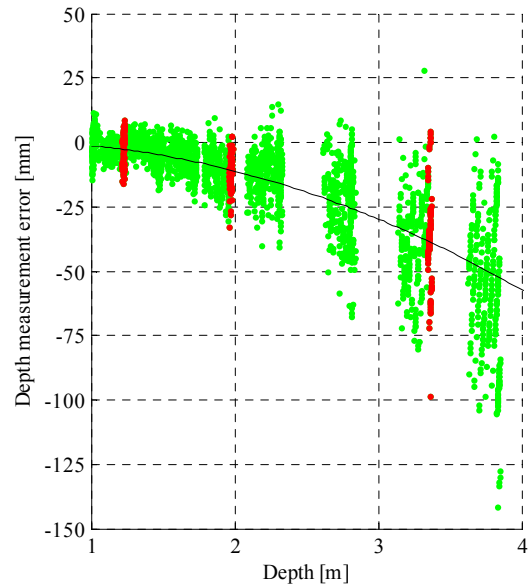


Figure 6. Depth measurement correction

It is important to underline that the error reduction that can be achieved by applying the described procedure is indeed a reduction of differences between reading of depth sensor and depth values obtained using the sensor's RGB camera. Therefore, well calibrated camera is an absolute prerequisite for a quality calibration.

#### 4. SENSOR ACCURACY

In this section, an analysis is made of achievable accuracy of both uncalibrated and calibrated sensor and the improvement introduced by calibration.



Figure 7. Views used in accuracy analysis

To this end, two possible scenarios are examined. In the first case, a use of only the depth sensor is assumed. An example of this scenario in the context of robotic SLAM (Simultaneous Localization and Mapping) provide works of Audras et al. [20] and Fioraio and Konolige [21] where the issue is on direct utilizing point clouds produced by Kinect to build a dense map of environment. Registration of the point clouds in successive depth frames is performed in an iterative process known as ICP (Iterative Closest Point). An alternative scenario, allowing to speed-up the relatively slow ICP process, involves the registration using sparse salient keypoints in 2D images obtained from the RGB

camera, whereas the depth of the keypoints is found by searching the depth map. Representative implementations of this approach are described by Henry et al. [22] and Endres et al. [23]. From the standpoint of Kinect accuracy, the main characteristic of this approach is a joint use of the depth sensor and the RGB camera and the concern is on errors of 3D coordinates inferred for selected points in RGB images.

In the framework of the analysis, three levels of sensor modeling are investigated in both examined scenarios: the nominal sensor model, the sensor with calibrated RGB/IR cameras, and the case with the additional depth measurement model calibration.

The analysis is based on measurements conducted on the same checkerboard employed in sensor calibration, using the views of the checkerboard shown in Fig. 7 (with the exclusion of the three central views in 2<sup>nd</sup>, 5<sup>th</sup>, and 8<sup>th</sup> row) that were used in calibration of the depth measurement model). As in the calibration case, measurement points correspond to inner 90 corners of the checkerboard.

For the purpose of the analysis, the views were divided into nine groups, each containing two or three views of the checkerboard placed approximately at the same position with respect to the sensor but in different orientations. In this manner, a set of nine clusters of measurement points was obtained. The clusters were used to estimate RMS errors and standard deviations for different distances from the sensor.

##### 4.1. Depth sensor accuracy

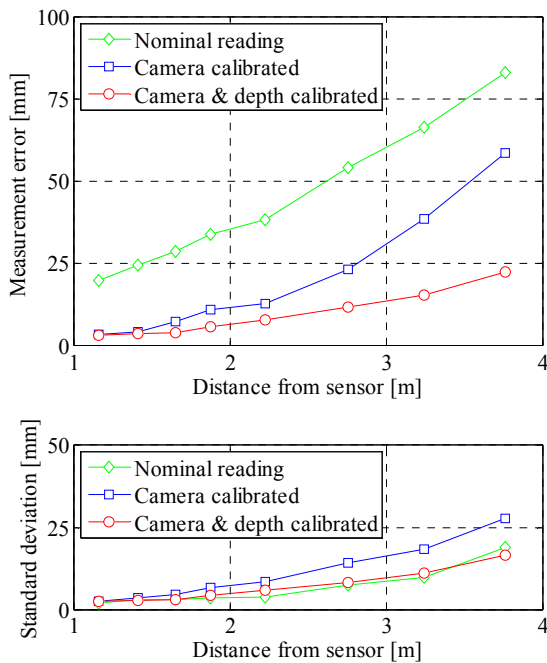
In this analysis, the accuracy of 3D coordinates  $\mathbf{X}_{IR}(i, j, k)$  determined by the sensor for selected points  $\mathbf{x}_{IR}^{(P)}(i, j, k)$  in its depth map is examined. The points  $\mathbf{x}_{IR}^{(P)}$  are selected as projections of checkerboard corners and they are computed using the same procedure as in depth measurement calibration: first, the transform  ${}^{RGB}\mathbf{T}_C(k)$  is determined for each view  $k$ , then (14) is applied to find the best guess  $\hat{\mathbf{X}}_{IR}(i, j, k)$  of actual external coordinates from which  $\mathbf{x}_{IR}^{(P)}(i, j, k)$  are calculated using (15). Sensor output is afterward generated by applying in order (16), (13), (3), and (4). (From the order of calculation it is seen that the factors affecting the accuracy are the IR camera and depth measurement algorithm.)

The resulting deviations of  $\mathbf{X}_{IR}(i, j, k)$  from  $\hat{\mathbf{X}}_{IR}(i, j, k)$  are clustered according to average depth and the obtained statistics (root mean square error  $\Delta L$  and standard deviation  $\sigma$ ) for different clusters is summarized in Table 2 and Fig. 8. It is seen that the application of nominal model produces large average errors, which are on the order of 35mm at the distance of 2m and on the order of 75mm at the distance of 3m. Calibration of the IR camera yields significant reduction of errors at shorter distances. (It could be noted here that almost the same results have been obtained after neglecting all deviations in parameters of the camera except for the focal length. This insensitivity is

attributed to small camera distortion and the fact that the measurement points were always in the central region of the image.) However, the errors remain large at distances larger than 2.5m. In this range, calibration of depth model allows decreasing average errors more than two times compared to the case of calibrating only camera model and more than three times compared to the nominal sensor model.

**Table 2. Depth sensor accuracy indicators**

Depth [m]	Nominal reading		With calibrated camera		With calibrated camera and depth model	
	$\Delta L$	$\sigma$	$\Delta L$	$\sigma$	$\Delta L$	$\sigma$
	[mm]	[mm]	[mm]	[mm]	[mm]	[mm]
0.96	17.1	2.7	4.7	3.6	4.7	3.4
1.16	19.9	2.1	3.5	2.4	3.2	2.4
1.41	24.6	3.0	4.2	3.5	3.6	2.7
1.65	28.8	3.0	7.4	4.5	4.0	3.0
1.88	33.9	3.5	10.9	6.6	5.7	4.2
2.23	38.2	3.9	12.7	8.4	7.8	5.9
2.76	54.2	7.4	23.3	14.1	11.7	8.1
3.24	66.3	9.7	38.6	18.3	15.5	11.0
3.76	83.0	19.0	58.6	27.8	22.4	16.6

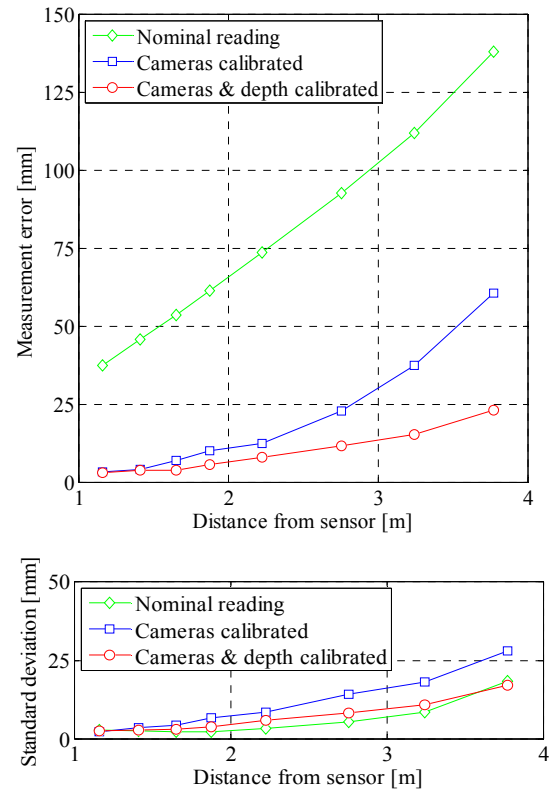


**Figure 8. Depth sensor accuracy**

#### 4.2. Joint RGB camera/depth sensor accuracy

In the second analyzed scenario, it is assumed that the inputs are pixel coordinates  $\mathbf{x}_{RGB}^{(P)}(i, j, k)$  of a point in RGB image for which it is requested to obtain the corresponding 3D coordinates  $\mathbf{X}_{IR}(i, j, k)$ . Compared to the first scenario, the only essential difference are unknown pixel coordinates  $\mathbf{x}_{IR}^{(P)}(i, j, k)$ . In this work, the task of finding coordinates  $\mathbf{x}_{IR}^{(P)}$  corresponding to  $\mathbf{x}_{RGB}^{(P)}$  was realized by using the considered (approximate) model of the RGB camera to transform  $\mathbf{x}_{RGB}^{(P)}$  into undistorted normalized coordinates  $\mathbf{x}_{RGB}$ , generating a ray through the center of the RGB camera

and the point  $\mathbf{x}_{RGB}$ , and finally finding an intersection of the ray through the undistorted depth map.



**Figure 9. Joint RGB camera/depth sensor accuracy**

As in the first scenario, points  $\mathbf{X}_{IR}(i, j, k)$  are further compared to  $\hat{\mathbf{X}}_{IR}(i, j, k)$  and the statistics obtained for clusters of points is calculated. The results are illustrated in Fig. 9, where it is seen that the errors obtained with nominal parameters enlarged by approximately 50% compared to the first scenario. On the other hand, errors obtained with calibrated RGB camera almost did not change, what is expected having in mind that the evaluation of  $\hat{\mathbf{X}}_{IR}(i, j, k)$  and  $\mathbf{X}_{IR}(i, j, k)$  was done using the same RGB camera model.

#### 5. CONCLUSION

This study confirmed large RMS errors of considered sensors. The errors were on the order of 35mm (standard deviation on the order of 10mm) at the distance of 2m and on the order of 75mm (standard deviation on the order of 15mm) at the distance of 3m. When considering joint use of the depth sensor with its associated RGB camera, the effective RMS errors enlarged by 50%.

Errors could be reduced by calibration of sensor's camera and depth measurement model. The procedure proposed in [5,6] proved to be effective as it allowed reducing RMS errors more than three times compared to the case when nominal sensor model was employed.

Calibration of depth model was an important element of the overall calibration as it allowed reducing by 50% the RMS errors left after calibration of sensor's cameras.

## ACKNOWLEDGMENT

This work was supported in part by the Serbian Ministry of Education, Science and Technology Development under projects III-44008 and TR-35003.

## REFERENCES

- [1] Microsoft Corporation: Kinect for Xbox 360, online at <http://www.microsoft.com/education/ww/products/Pages/kinect.aspx>
- [2] Microsoft Corporation: Kinect for Windows, online at <http://www.microsoft.com/en-us/kinectforwindows/>
- [3] ASUSTeK: WAVI Xtion, Intuitive living room experience, online at <http://event.asus.com/wavi/product/xtion.aspx>
- [4] PrimeSense: Get your sensor, online at <http://www.primesense.com/developers/get-your-sensor/>
- [5] Karan, B.: Accuracy improvements of consumer-grade 3D sensors for robotic applications, in Proc. IEEE 11th Int. Symp. Intelligent Systems and Informatics (SISY), Subotica, Serbia, Sept. 2013, pp. 141-146.
- [6] Karan, B.: Calibration of depth measurement model for Kinect-type 3D vision sensors, in Proc. 21st Int. Conf. Computer Graphics, Visualization and Computer Vision (WSCG 2013), Plzen, Czech Republic, June 2013, pp. 61-64.
- [7] Dutta, T.: Evaluation of the Kinect sensor for 3-D kinematic measurement in the workplace, in Applied Ergonomics, Vol. 43, No. 4, pp. 645-649, 2012.
- [8] Gonzalez-Jorge, H., Riveiro, B., Vazquez-Fernandez, E., et al.: Metrological evaluation of Microsoft Kinect and Asus Xtion sensors, in Measurement: Journal of the International Measurement Confederation, Vol. 46, No. 6, pp. 1800-1806, 2013.
- [9] Burrus, N.: Kinect calibration, online at <http://nicolas.burrus.name/index.php/Research/KinectCalibration>
- [10] Zhang, C. and Zhang, Z.: Calibration between depth and color sensors for commodity depth cameras, in Proc. 2011 IEEE Int. Conf. Multimedia and Expo (ICME), Barcelona, Spain, July 2011, pp. 1-6.
- [11] Khoshelham, K. and Elberink, S.O.: Accuracy and resolution of Kinect depth data for indoor mapping applications, in Sensors, Vol. 12, pp. 1437-1454, 2012.
- [12] Smisek, J., Jancosek, M. and Pajdla, T.: 3D with Kinect, in Proc. 2011 IEEE Int. Conf. Computer Vision Workshops, Barcelona, Spain, Nov. 2011, pp. 1154-1160.
- [13] Herrera C., D., Kannala, J. and Heikkilä, J.: Joint depth and color camera calibration with distortion correction, in IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 34, No. 10, pp. 2058-2064, Oct. 2012.
- [14] OpenNI Consortium: OpenNI, the standard framework for 3D sensing, online at <http://www.openni.org/>
- [15] Microsoft Corporation: Kinect for Windows SDK, online at <http://msdn.microsoft.com/en-us/library/hh855347.aspx>
- [16] Draelos, M., Deshpande, N. and Grant, E.: The Kinect up close: Adaptations for short-range imaging, in Proc. 2012 IEEE Conf. Multisensor Fusion and Integration for Intelligent Systems (MFI), Hamburg, Germany, Sept. 2012, pp. 251-256.
- [17] Geiger, A., Moosmann, F., Car, Ö. et al.: Automatic camera and range sensor calibration using a single shot, in Proc. 2012 IEEE Int. Conf. Robotics and Automation (ICRA), Saint Paul, Minnesota, USA, May 2012, pp. 3936-3943.
- [18] Shibo L. and Qing, Z.: A new approach to calibrate range image and color image from Kinect, in Proc. 4. Int. Conf. Intelligent Human-Machine Systems and Cybernetics (IHMSC), Nanchang, China, Aug. 2012, Vol. 2, pp. 252-255.
- [19] Bouguet, J.-Y.: Camera Calibration Toolbox for Matlab, online at [http://www.vision.caltech.edu/bouguetj/calib\\_doc](http://www.vision.caltech.edu/bouguetj/calib_doc)
- [20] Audras, C. Comport, A.I., Meilland, M. and Rives, P.: Real-time dense appearance-based SLAM for RGB-D sensors, in Proc. Australasian Conf. Robotics and Automation, Melbourne, Australia, Dec. 2011.
- [21] Fioraio, N. and Konolige, K.: Realtime visual and point cloud SLAM, in Proc. 2nd Workshop on RGB-D: Advanced Reasoning with Depth Cameras, Robotics: Science and Systems VII Conference, Los Angeles, USA, June 2011.
- [22] Henry, P. Krainin, M., Herbst, E. et al.: RGB-D mapping: Using depth cameras for dense 3D modeling of indoor environments, Springer Tracts in Advanced Robotics, Vol. 79, Springer Berlin Heidelberg, 2014, pp. 477-491.
- [23] Endres, F. Hess, J., Engelhard, N. et al.: An evaluation of the RGB-D SLAM system, Proc. 2012 IEEE Int. Conf. Robotics and Automation (ICRA), IEEE, 2012, pp. 1691-1696.

---

## КАЛИБРАЦИЈА RGB-D СЕНЗОРА ТИПА КИНЕКТ ЗА РОБОТИЧАРСКЕ ПРИМЕНЕ

Бранко Каран

У раду је представљен калибрациони модел погодан за софтверску калибрацију RGB-D сензора типа Кинект. Додатно, описана је двокорачна калибрациона процедура која претпоставља коришћење само простог шаблона у виду шаховске табле. Коначно, у раду је приказана калибрациона студија у којој је показано да тачност сензора може да се калибрацијом побољша 3 до 5 пута. Резултати добијени у овој студији коришћењем калибрационих модела различите сложености откривају да је корекција мерења дубине важна компонента калибрације пошто је њоме могуће да се за 50% смање грешке у читавању сензора.