Depth error correction for projector-camera based consumer depth cameras

Hirotake Yamazoe[1](✉), Hiroshi Habe[2], Ikuhisa Mitsugami[3], and Yasushi Yagi[4]

Abstract This paper proposes a depth measurement error model for consumer depth cameras such as the Microsoft Kinect, and a corresponding calibration method. These devices were originally designed as video game interfaces, and their output depth maps usually lack sufficient accuracy for 3D measurement. Models have been proposed to reduce these depth errors, but they only consider camera-related causes. Since the depth sensors are based on projector-camera systems, we should also consider projector-related causes. Also, previous models require disparity observations, which are usually not output by such sensors, so cannot be employed in practice. We give an alternative error model for projector-camera based consumer depth cameras, based on their depth measurement algorithm, and intrinsic parameters of the camera and the projector; it does not need disparity values. We also give a corresponding new parameter estimation method which simply needs observation of a planar board. Our calibrated error model allows use of a consumer depth sensor as a 3D measuring device. Experimental results show the validity and effectiveness of the error model and calibration procedure.

Keywords consumer depth camera; intrinsic calibration; projector; distortion

1 College of Information Science and Engineering, Ritsumeikan University, Shiga, 525-8577, Japan. E-mail: yamazoe@fc.ritsumei.ac.jp (✉).

2 Faculty of Science and Engineering, Kindai University, Osaka, 577-8502, Japan. E-mail: habe@kindai.ac.jp.

3 Graduate School of Information Sciences, Hiroshima City University, Hiroshima, 731-3194, Japan. E-mail: mitsugami@hiroshima-cu.ac.jp.

4 The Institute of Scientific and Industrial Research, Osaka University, Osaka, 567-0047, Japan. E-mail: yagi@am.sanken.osaka-u.ac.jp.

1 Introduction

Recently, various consumer depth cameras such as the Microsoft Kinect V1/V2, Asus Xtion, etc. have been released. Since such consumer depth sensors are inexpensive and easy to use, these devices are widely deployed in various fields for a wide variety of applications [1, 2].

These consumer depth cameras can be divided into two categories: (i) projector-camera based systems in which a projector casts a structured pattern onto the surface of a target object, and (ii) time-of-flight (ToF) sensors that measure the time taken for light to travel from a source to an object and back to a sensor. ToF sensors generally give more accurate depths than projector-camera based ones, which are, however, still useful because of their simplicity and low cost.

Since projector-camera based devices include cameras and a projector, output errors may be caused in errors in determining the intrinsic parameters. As long as such devices are used as human interfaces for video games, such errors are unimportant. For example, even when a Kinect V1 captures a planar object, the resultant depth maps have errors (see Fig. 1, left) as also reported elsewhere [3, 4]. Thus, in this paper, we focus on projector-camera based consumer depth cameras and propose a depth error correction method based on their depth measurement algorithm. Various intrinsic calibration methods have already been proposed for Kinect and other projector-camera based depth cameras [3–11]. Smisek et al. [3] and Herrera et al. [4] proposed calibration and depth correction methods for Kinect that reduce the depth observation errors. Raposo et al. [6] extended Herrera et al.'s method to improve stability and
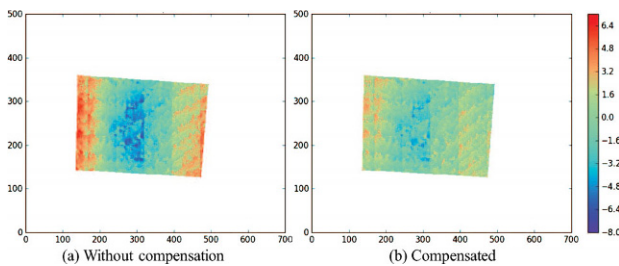
Fig. 1 Left: observation errors in Kinect output. Right: compensated values using our method.

speed. However, their methods only considered distortion due to the infrared (IR) cameras. Since projector-camera based depth sensors include cameras and a projector, we should also consider projector-related sources of error.

We previously proposed a depth error model for Kinect including projector-related distortion [5]. Darwish et al. [8] also proposed a calibration algorithm that considers both camera and projector-related parameters for Kinect. However, these methods as well as other previous methods require disparity observations, and these are not generally provided by such sensors. Thus, methods that require disparity observations cannot be employed in practice for error compensation for data from existing commercial sensors.

Some researchers employ non-parametric models for depth correction but a calibration board needs to be shown perpendicular to the sensor [9, 11], or ground truth data obtained by simultaneous localization and mapping (SLAM) are required [12, 13]. Jin et al. [10] proposed a calibration method using cuboids, but, their method is also based on disparity observations. Other researchers proposed error distribution models for Kinect [14, 15], but this research did not focus on error compensation.

To provide straightforward procedures for calibration and error compensation for depth data, including previously captured data, our method introduces a parametric error model that considers (i) both camera and projector distortion, and (ii) errors in the parameters used to convert disparity observations to actual disparity. To estimate the parameters in the error model, we propose a simple method that resembles the common color camera calibration method [16]. Having placed a planar calibration board in front of the depth camera and captured a set of images, our method efficiently

optimizes the parameters allowing us to reduce the depth measurement errors (see Fig. 1, right). Our compensation model only requires depth data, without the need for disparity observations. Thus we can apply our error compensation to any depth data captured by projector-camera based depth cameras.

We note that the calibration method introduced in this paper is designed for Kinect because it is the most common projector-camera based depth sensor. However, it potentially generally more useful because it is based on a principle common to other projector-camera based depth sensors.

Section 2 describes the measurement algorithm used by Kinect, and Section 3 describes our parametric error model and parameter estimation. Section 4 shows experimental results demonstrating the effectiveness of our proposed method, while Section 5 summarizes our paper.

2 Depth error model

2.1 Depth measurement by Kinect

Since our method is based on the measurement algorithm used by the Kinect, we first outline this algorithm and this depth sensor, which consists of an IR camera and an IR projector. The IR projector projects special fixed patterns (speckle patterns) on the target observed by the IR camera. By comparing the observed and reference patterns captured in advance, Kinect estimates depth information for the target. The reference patterns are observations made by the IR camera when the IR projector casts the speckle pattern on the reference plane Π_0 [17] (see Fig. 2).

Here, we assume pattern $P(\boldsymbol{x}_{\mathrm{p}i})$ is projected in the direction of point $\boldsymbol{x}_{\mathrm{p}i} = [x_{\mathrm{p}i}, y_{\mathrm{p}i}]^{\mathrm{T}}$ onto the reference plane Π_0, and pattern $P(\boldsymbol{x}_{\mathrm{p}i})$ on Π_0 is projected onto the 2D position $\boldsymbol{x}_{\mathrm{c}i}^{(\Pi_0)} = [x_{\mathrm{c}i}^{(\Pi_0)}, y_{\mathrm{c}i}^{(\Pi_0)}]^{\mathrm{T}}$ for the IR camera. We obtain the following relationship:

$$x_{\mathrm{c}i}^{(\Pi_0)} = x_{\mathrm{p}i} + fw/Z_0 \qquad (1)$$

where w is the baseline distance between the camera and the projector, f is the focal length of the IR camera (and the IR projector), and Z_0 is the distance between the reference plane Π_0 and the Kinect.

Next, we consider the target observation measured at point Q_i and assume that pattern $P(\boldsymbol{x}_{\mathrm{p}i})$ is observed at $\boldsymbol{x}_{\mathrm{c}i}$ using the IR camera. By considering
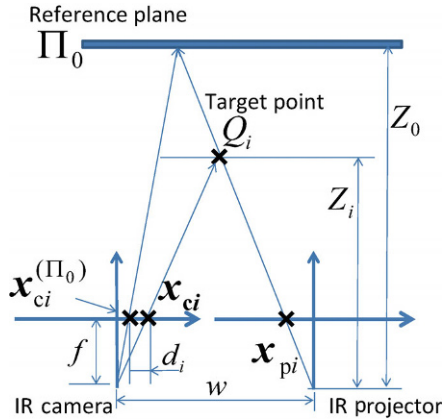
Fig. 2 Depth measurement by Kinect.

the reference patterns, the pattern's observed position when it is projected onto reference plane Π_0, $\boldsymbol{x}_{\mathrm{c}i}^{(\Pi_0)}$, can be obtained. We calculate disparity δ_i from the reference plane observation at $\boldsymbol{x}_{\mathrm{c}i}$ as follows:

$$\delta_i = x_{\mathrm{c}i} - x_{\mathrm{c}i}^{(\Pi_0)} = x_{\mathrm{c}i} - x_{\mathrm{p}i} - fw/Z_0 \qquad (2)$$

Then $\boldsymbol{X}_i$, which is the 3D position of point Q_i, can be calculated as

$$\boldsymbol{X}_i = \left[\frac{(x_{\mathrm{c}i} - x_{\mathrm{cc}})Z_i}{f}, \frac{(y_{\mathrm{c}i} - y_{\mathrm{cc}})Z_i}{f}, \frac{fw}{fw/Z_0 + \delta_i} \right]^{\mathrm{T}} \qquad (3)$$

where x_{cc} and y_{cc} are the IR camera's principal points and Z_i is the depth of point Q_i.

Kinect does not output disparity values, but only normalized observations δ_i' from 0 to 2047 (in *Kinect disparity units*: kdu) [17], where $\delta_i = m\delta_i' + n$. The driver software for Kinect (Kinect for Windows SDK and OpenNI) uses these to calculate and output depth values Z_i based on the following equation:

$$Z_i = \frac{fw}{fw/Z_0 + m\delta_i' + n} = \frac{fw}{d_i} \qquad (4)$$

The disparity between the camera and projector d_i can be expressed as follows:

$$d_i = fw/Z_0 + m\delta_i' + n \qquad (5)$$

Note that recent versions of the driver software do not support output of disparities δ_i', so these are generally unobtainable. Instead, we propose a method to calibrate and compensate the depth data obtained by Kinect that does not require either the disparity or normalized disparity observations.

2.2 Depth error model

The depth measurement model described above holds only in an ideal case. In practice, when Kinect observes a planar target, the output depth maps have errors, as previously noted [3, 4] (see Fig. 1). To be able to compensate for them, we consider not only camera distortion but also the projector distortion in our model.

2.2.1 Distortion parameters

A well-known lens distortion model is

$$\breve{\boldsymbol{x}}_{\mathrm{c}i} =$$
$$\begin{bmatrix} x_{\mathrm{c}i} + (x_{\mathrm{c}i} - x_{\mathrm{cc}}) \left[k_{\mathrm{c}1} \boldsymbol{u}_{\mathrm{c}i}^{\mathrm{T}} \boldsymbol{u}_{\mathrm{c}i} + k_{\mathrm{c}2} (\boldsymbol{u}_{\mathrm{c}i}^{\mathrm{T}} \boldsymbol{u}_{\mathrm{c}i})^2 \right] \\ y_{\mathrm{c}i} + (y_{\mathrm{c}i} - y_{\mathrm{cc}}) \left[k_{\mathrm{c}1} \boldsymbol{u}_{\mathrm{c}i}^{\mathrm{T}} \boldsymbol{u}_{\mathrm{c}i} + k_{\mathrm{c}2} (\boldsymbol{u}_{\mathrm{c}i}^{\mathrm{T}} \boldsymbol{u}_{\mathrm{c}i})^2 \right] \end{bmatrix}$$
$$(6)$$

where $\boldsymbol{x}_{\mathrm{c}i} = [x_{\mathrm{c}i}, y_{\mathrm{c}i}]^{\mathrm{T}}$ and $\breve{\boldsymbol{x}}_{\mathrm{c}i}$ are the ideal and observed, distorted 2D positions, and $\boldsymbol{u}_{\mathrm{c}i}$ gives the normalized coordinates of $\boldsymbol{x}_{\mathrm{c}i}$. $k_{\mathrm{c}1}$ and $k_{\mathrm{c}2}$ are the distortion parameters of the IR camera.

We assume the same distortion model can be used for the projector:

$$\breve{\boldsymbol{x}}_{\mathrm{p}i} =$$
$$\begin{bmatrix} x_{\mathrm{p}i} + (x_{\mathrm{p}i} - x_{\mathrm{pc}}) \left[k_{\mathrm{p}1} \boldsymbol{u}_{\mathrm{p}i}^{\mathrm{T}} \boldsymbol{u}_{\mathrm{p}i} + k_{\mathrm{p}2} (\boldsymbol{u}_{\mathrm{p}i}^{\mathrm{T}} \boldsymbol{u}_{\mathrm{p}i})^2 \right] \\ y_{\mathrm{p}i} + (y_{\mathrm{p}i} - y_{\mathrm{pc}}) \left[k_{\mathrm{p}1} \boldsymbol{u}_{\mathrm{p}i}^{\mathrm{T}} \boldsymbol{u}_{\mathrm{p}i} + k_{\mathrm{p}2} (\boldsymbol{u}_{\mathrm{p}i}^{\mathrm{T}} \boldsymbol{u}_{\mathrm{p}i})^2 \right] \end{bmatrix}$$
$$(7)$$

where $\boldsymbol{x}_{\mathrm{p}i}$ and $\breve{\boldsymbol{x}}_{\mathrm{p}i}$ are the ideal and distorted 2D positions, and $\boldsymbol{u}_{\mathrm{p}i}$ gives the normalized coordinates of $\boldsymbol{x}_{\mathrm{p}i}$. $[x_{\mathrm{pc}}, y_{\mathrm{pc}}]^{\mathrm{T}}$ is the principal point of the projector, and $k_{\mathrm{p}1}$ and $k_{\mathrm{p}2}$ are the distortion parameters of the IR projector.

We now consider pattern $P(\boldsymbol{x}_{\mathrm{p}i})$ projected in the direction of point $\boldsymbol{x}_{\mathrm{p}i}$ (see Fig. 3). However, because of projector distortion, pattern $P(\boldsymbol{x}_{\mathrm{p}i})$ is actually projected in the direction of point $\breve{\boldsymbol{x}}_{\mathrm{p}i}$, and is projected onto point Q_i'. In the camera, pattern $P(\boldsymbol{x}_{\mathrm{p}i})$ is actually projected onto position $\breve{\boldsymbol{x}}_{\mathrm{c}i}$ because of the camera distortion. Let $\breve{d}_i$ be the observed (distorted) disparity at $\breve{\boldsymbol{x}}_{\mathrm{c}i}$ and

$$\breve{d}_i = \breve{x}_{\mathrm{c}i} - x_{\mathrm{p}i} \qquad (8)$$

On the other hand, considering Q_i' in Fig. 3, the ideal disparity d_i corresponding to point Q_i' should be

$$d_i = x_{\mathrm{c}i} - \breve{x}_{\mathrm{p}i} = \breve{d}_i - \epsilon_{\mathrm{c}} - \epsilon_{\mathrm{p}} \qquad (9)$$

where $\epsilon_{\mathrm{c}} = \breve{x}_{\mathrm{c}i} - x_{\mathrm{c}i}$ and $\epsilon_{\mathrm{p}} = \breve{x}_{\mathrm{p}i} - x_{\mathrm{p}i}$.

2.2.2 Proposed error model

Equation (9) expresses the relation between the ideal disparity d_i and the observed disparity $\breve{d}_i$. In practice, since the parameters in Eq. (5), i.e., f, w, Z_0, m, and n include errors, we need to
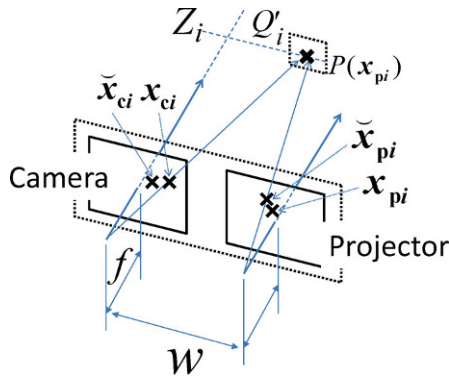
Fig. 3 Proposed error model.

compensate for them. Here, let $\breve{d}_i'$ and $\breve{d}_i$ be the ideal values and the values calculated by Eq. (5) based on the error parameters and the observed disparity. Considering the errors in the parameters in Eq. (5) and collecting the coefficients, the following relations can be obtained:

$$\breve{d}_i' = \alpha \breve{d}_i + \beta \tag{10}$$

where α and β are parameters for compensating errors in f, w, Z_0, m, and n. A detailed derivation of Eq. (10) is shown in Appendix A. Thus, the ideal disparity d_i can be expressed as follows:

$$\begin{aligned} d_i &= \breve{d}_i' - \epsilon_{\mathrm{c}} - \epsilon_{\mathrm{p}} \\ &= (\alpha \breve{d}_i + \beta) - \epsilon_{\mathrm{c}} - \epsilon_{\mathrm{p}} \end{aligned} \tag{11}$$

By introducing α and β, we can compensate for errors in the parameters in Eq. (5) without observing the normalized disparity itself. Therefore, we calibrate not only the distortion parameters of the camera and the projector but also α and β allowing us to compensate for errors in these values. In the next section, we describe parameter estimation for this error model.

3 Algorithm

3.1 Overview

In consumer depth sensors, since the projection patterns cannot be controlled, we cannot directly estimate the projector's distortion parameters. Instead, we estimate the error model parameters using the process flow shown in Fig. 4.

First, we obtain N IR images and corresponding depth data for a calibration board (of known size and pattern) in arbitrary poses and positions. This lets us perform intrinsic calibration of the IR camera by
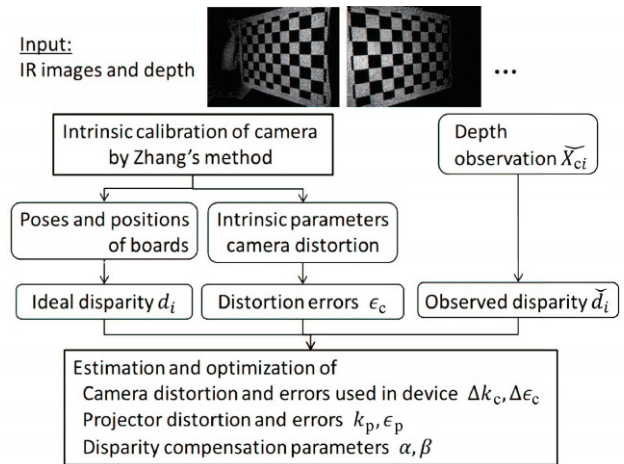

Fig. 4 Process flow.

Zhang's method [16]. As described in the previous section, we model the depth errors based on Eq. (11), and ideal disparity d_i and observed disparity $\breve{d}_i$ are required. Here, we assume that the poses and positions of the board estimated by intrinsic camera calibration are ideal depth values, and calculate ideal disparity d_i from these poses and positions. The observed disparity values can be calculated from the observed depth values. Next we estimate the error model parameters by minimizing Eq. (11) based on d_i and $\breve{d}_i$. Table 1 summarizes the notation used in the following.

3.2 Camera calibration

First, intrinsic calibration of the IR camera is performed using the N images captured by the IR cameras, using Zhang's method [16]. For camera calibration, X_{k}, $x_{\mathrm{bk}}^{(j)}$, the size of the chessboard, and the number of checker patterns on the chessboard should be given. Zhang's method can estimate the focal length (f), the principal point (u_{cc}, v_{cc}), and the camera distortion parameters ($k_{\mathrm{c}} = \{k_{\mathrm{c1}}, k_{\mathrm{c2}}\}$). The disparity differences caused by the camera lens

Table 1 Notation

i	index of points
j	index of observations, $j = 1, \ldots, N$
k	index of chessboard corner points
$\breve{\boldsymbol{x}}_{ci}^{(j)}$	2D positions in image j
$\breve{\boldsymbol{X}}_{ci}^{(j)}$	3D positions at $\breve{\boldsymbol{x}}_{ci}^{(j)}$ obtained by sensor
$\boldsymbol{X}_{\mathrm{k}}$	3D positions of chessboard corners ($Z_{\mathrm{bk}} = 0$)
$\boldsymbol{x}_{\mathrm{bk}}^{(j)}$	2D positions of corners in image j
$\breve{\boldsymbol{X}}_{\mathrm{bk}}^{(j)}$	3D positions at $\boldsymbol{x}_{\mathrm{bk}}^{(j)}$ obtained by sensor

distortion may be considered. Let $\boldsymbol{k}'_{\mathrm{c}}$ be camera the distortion parameter, and ϵ'_{c} be disparity error caused by $\boldsymbol{k}'_{\mathrm{c}}$. Then $\boldsymbol{k}_{\mathrm{c}}$ and ϵ_{c} can be expressed as follows:

$$\begin{cases} \boldsymbol{k}_{\mathrm{c}} = \boldsymbol{k}'_{\mathrm{c}} + \Delta\boldsymbol{k}_{\mathrm{c}} \\ \epsilon_{\mathrm{c}} = \epsilon'_{\mathrm{c}} + \Delta\epsilon_{\mathrm{c}} \end{cases} \quad (12)$$

and Eq. (11) can be rewritten as follows:

$$d_i = (\alpha \breve{d}_i + \beta) - \Delta\epsilon_{\mathrm{c}} - \epsilon_{\mathrm{p}} \quad (13)$$

In addition, we can obtain the board's poses and positions in each image j: $(\boldsymbol{R}^{(j)}, \boldsymbol{t}^{(j)})$. This information is used in the following processes.

3.3 Projector calibration and disparity compensation parameter estimation

Next, we estimate the distortion parameters for the projector and the disparity conversion parameters. To do so, we use the relations in Eq. (13) to give the following equation:

$$\sum_j \sum_i \{d_i^{(j)} - (\alpha\breve{d}_i^{(j)} + \beta - \Delta\epsilon_{\mathrm{c}i}^{(j)} - \epsilon_{\mathrm{p}i}^{(j)})\} = 0 \quad (14)$$

The ideal disparities $d_i^{(j)}$ can be calculated from the estimated poses and positions $\boldsymbol{R}^{(j)}$, $\boldsymbol{t}^{(j)}$ as follows:

$$d_i^{(j)} = \frac{wf}{Z_{\mathrm{b}}(\boldsymbol{x}_{\mathrm{c}i}^{(j)})} \quad (15)$$

where $\boldsymbol{X}_{\mathrm{b}}^{(j)}(\boldsymbol{x}_{\mathrm{c}i}^{(j)})$ gives the 3D position on the board corresponding to direction $\boldsymbol{x}_i^{(j)}$ (see Fig. 5), and $Z_{\mathrm{b}}^{(j)}(\boldsymbol{x}_{\mathrm{c}i}^{(j)})$ is the Z component of $\boldsymbol{X}_{\mathrm{b}}^{(j)}(\boldsymbol{x}_{\mathrm{c}i}^{(j)})$.

The observed disparity values $\breve{d}_i$ can be estimated using:

$$\breve{d}_i^{(j)} = \frac{wf_{\mathrm{o}}}{\breve{Z}_{\mathrm{c}i}^{(j)}} \quad (16)$$
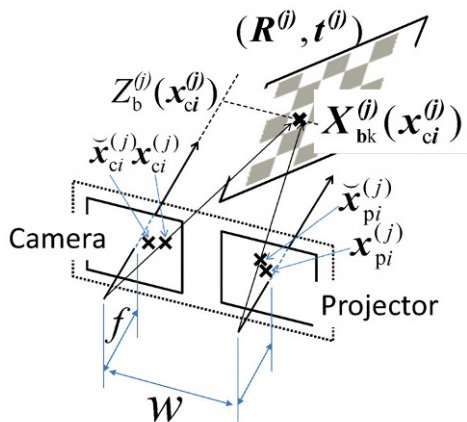
Fig. 5 Relationship between 2D and 3D observations.

where $\breve{Z}_{\mathrm{c}i}^{(j)}$ is the observed depth value at position $\boldsymbol{x}_{\mathrm{c}i}^{(j)}$. f_{o} is the focal length for the depth sensor used for calculating observed disparity values $\breve{d}_i^{(j)}$, and can be estimated from $\breve{\boldsymbol{X}}_{\mathrm{c}i}^{(j)}$.

$\epsilon_{\mathrm{c}i}^{(j)}$ can be expressed as follows:

$$\epsilon_{\mathrm{c}i}^{(j)} = \breve{x}_{\mathrm{c}i}^{(j)} - x_{\mathrm{c}i}^{(j)}$$
$$= -(\breve{x}_i^{(j)} - x_{\mathrm{cc}})\left[k_{\mathrm{c}1}(\breve{\boldsymbol{u}}_{\mathrm{c}i}^{(j)})^{\mathrm{T}}\breve{\boldsymbol{u}}_{\mathrm{c}i}^{(j)} + k_{\mathrm{c}2}((\breve{\boldsymbol{u}}_{\mathrm{c}i}^{(j)})^{\mathrm{T}}\breve{\boldsymbol{u}}_{\mathrm{c}i}^{(j)})^2\right]$$

$$(17)$$

where

$$\boldsymbol{x}_{\mathrm{c}i}^{(j)} =$$
$$\breve{\boldsymbol{x}}_{\mathrm{c}i}^{(j)} - \left(\breve{\boldsymbol{x}}_{\mathrm{c}i}^{(j)} - \begin{bmatrix} x_{\mathrm{cc}} \\ y_{\mathrm{cc}} \end{bmatrix}\right)\left[k_{\mathrm{c}1}(\breve{\boldsymbol{u}}_{\mathrm{c}i}^{(j)})^{\mathrm{T}}\breve{\boldsymbol{u}}_{\mathrm{c}i}^{(j)} + k_{\mathrm{c}2}((\breve{\boldsymbol{u}}_{\mathrm{c}i}^{(j)})^{\mathrm{T}}\breve{\boldsymbol{u}}_{\mathrm{c}i}^{(j)})^2\right]$$

$$(18)$$

and we employ the approximate undistorted model [18].

Based on the above equations, we can estimate $\Delta\boldsymbol{k}_{\mathrm{c}}$, $\boldsymbol{k}_{\mathrm{p}}$, α, and β by minimization as below:

$$\left[\hat{k}_{\mathrm{p}}, \hat{\alpha}, \hat{\beta}\right] =$$
$$\arg\min_{\Delta k_{\mathrm{c}}, k_{\mathrm{p}}, \alpha, \beta} \sum_j \sum_i \{d_i^{(j)} - (\alpha\breve{d}_i^{(j)} + \beta - \Delta\epsilon_{\mathrm{c}i}^{(j)} - \epsilon_{\mathrm{p}i}^{(j)})\}$$

$$(19)$$

In our experiments, we employed the multi-start algorithm in the MATLAB and Global Optimization Toolbox (version 2017b) for optimization. Since it requires initial values, they were determined as follows. Note that w is not included in the parameters to be calibrated, because w and disparities d_i and $\breve{d}_i$ are proportional. Instead, we employ $w = 75$ mm based on Ref. [19].

We first consider the initial values of conversion parameters $\hat{\alpha}$ and $\hat{\beta}$. From Eqs. (15) and (16), we obtain the following relations by substituting 0 for $\epsilon_{\mathrm{c}i}^{(j)}$ and $\epsilon_{\mathrm{p}i}^{(j)}$ into Eq. (11):

$$d_i = \hat{\alpha}\breve{d}_i + \hat{\beta} \quad (20)$$

allowing $\hat{\alpha}$ and $\hat{\beta}$ to be determined by least-squares fitting.

Next, we consider the initial values for distortion errors $\hat{\epsilon}_{\mathrm{p}i}^{(j)}$. Considering the relations between the camera and the projector and Eq. (8), we can estimate $\boldsymbol{x}_{\mathrm{p}i}^{(j)}$ and $\breve{\boldsymbol{x}}_{\mathrm{p}i}^{(j)}$ as follows:

$$m\begin{bmatrix} \breve{\boldsymbol{x}}_{\mathrm{p}i}^{(j)} \\ 1 \end{bmatrix} = \boldsymbol{A}_{\mathrm{p}}\left(\boldsymbol{X}_{\mathrm{b}}^{(j)} - \begin{bmatrix} w \\ 0 \\ 0 \end{bmatrix}\right) \quad (21)$$

$$\boldsymbol{x}_{\mathrm{p}i}^{(j)} = \breve{\boldsymbol{x}}_{\mathrm{c}i}^{(c)} - \begin{bmatrix} \alpha\breve{\boldsymbol{d}}_i + \beta \\ 0 \end{bmatrix} \quad (22)$$

where $\boldsymbol{A}_{\mathrm{p}}$ are intrinsic parameters of the projector. Using Eq. (7), we can obtain the following equation:

$$\hat{\epsilon}_{\mathrm{p}i}^{(j)} = \breve{x}_{\mathrm{p}i}^{(j)} - x_{\mathrm{p}i}^{(j)}$$
$$= -\left(\breve{x}_{\mathrm{p}i}^{(j)} - \begin{bmatrix} x_{\mathrm{pc}} \\ y_{\mathrm{pc}} \end{bmatrix}\right)\left[\hat{k}_{\mathrm{p}1}(\breve{\boldsymbol{u}}_{\mathrm{p}i}^{(j)})^{\mathrm{T}}\breve{\boldsymbol{u}}_{\mathrm{p}i}^{(j)} + \hat{k}_{\mathrm{p}2}((\breve{\boldsymbol{u}}_{\mathrm{p}i}^{(j)})^{\mathrm{T}}\breve{\boldsymbol{u}}_{\mathrm{p}i}^{(j)})^2\right]$$
(23)

We then estimate the optimal values of $\epsilon_{\mathrm{p}i}$ (and $k_{\mathrm{p}1}, k_{\mathrm{p}2}$), α, and β by minimizing Eq. (19) from these initial values.

3.4 Depth compensation using calibration results

Finally, we describe the compensation process for the depth data obtained from the depth sensors.

Here, we consider the 3D data $\breve{\boldsymbol{X}}_{\mathrm{c}i}^{(j)}$ at pixel $\breve{\boldsymbol{x}}_{\mathrm{c}i}^{(j)}$. First, we obtain $\breve{d}_i$ and ϵ_{c} from Eqs. (16) and (17). $\breve{x}_{\mathrm{p}i}$ can be calculated from Eq. (22). Next, we calculate $\epsilon_{\mathrm{p}i}$ from Eq. (23). Then, the compensated disparity d_i can be calculated from Eq. (11), so the compensated depth Z_i can be obtained as

$$Z_i = \frac{fw}{d_i}$$
(24)

and the compensated 3D data $\boldsymbol{X}_i$ is given by

$$\boldsymbol{X}_i = \left[\frac{(x_{\mathrm{c}i} - x_{\mathrm{cc}})Z_i}{f}, \frac{(y_{\mathrm{c}i} - y_{\mathrm{cc}})Z_i}{f}, Z_i\right]^{\mathrm{T}}$$
(25)

4 Experiments

We performed the following experiments to confirm the validity of our proposed error model and error compensation. In the experiment, we used a Kinect for Xbox (Device 1, abbreviated Dev.1, etc.), a Kinect for Windows (Device 2), and an ASUS Xtion Pro (Device 3); all of these devices are based on the same Primesense measurement algorithm [20]. We compared the compensated results using the following three models and the observed raw data:
(a) our proposed method;
(b) model considering camera distortion and conversion parameters (without ϵ_{p});
(c) model considering only camera distortion errors (with $\Delta\epsilon_{\mathrm{c}}$);
(d) no compensation, i.e., observed raw data.

We captured 12 observations of the chessboard in different arbitrary poses and positions in the experiments. The distances between the board and the device were about 500–1300 mm. A leave-one-out method was used for evaluating the validity of the proposed error model: one observation was

used for evaluation and the remaining observations were used for estimating error model parameters. From the observations, we manually obtained the 2D positions of the chessboard corners (54 points per image).

Table 2 shows the residual errors after the calibration phase, and Table 3 shows the errors in evaluations. Here, the errors were calculated as the averaged distances between the compensated (or observed) positions and ground truth positions of the chessboard corners. We used the 3D positions obtained from the color camera observations as the ground truth positions.

These comparative results show that all three models can reduce errors compared to the uncompensated results, in both the calibration and evaluation phases. The errors compensated by (a) our proposed model were the lowest, followed by (b) the model that considered camera distortion and linear relations, and then (c) the model that considered only camera distortion. The number of parameters used in these models also has the same ordering: (a) has the most, followed by (b) and then (c). These results suggest that using all parameters considered in our proposed error model are helpful in improving the quality of the 3D depth data.

After calibration, we evaluated the flatness of the compensated observations for the chessboard, measuring plane fitting errors within the chessboard regions. Table 4 shows comparative results for these plane fitting errors.

These results show that the plane fitting errors in compensated observations from our proposed model

Table 2 Comparison of averaged errors during calibration

	(a) proposed (mm)	(b) w/o ϵ_{p} (mm)	(c) w ϵ_{c} (mm)	(d) w/o comp. (mm)
Dev.1	2.56	2.79	3.88	10.86
Dev.2	2.20	2.60	6.28	6.98
Dev.3	1.33	1.47	2.33	2.26

Table 3 Comparison of averaged errors in evaluation

	(a) proposed (mm)	(b) w/o ϵ_{p} (mm)	(c) w ϵ_{c} (mm)	(d) w/o comp. (mm)
Dev.1	4.36	3.99	4.56	10.13
Dev.2	3.46	3.72	4.59	5.56
Dev.3	1.65	1.75	2.15	2.14

Table 4 Comparison of plane fitting errors in evaluation

	(a) proposed (mm)	(b) w/o ϵ_P (mm)	(c) w ϵ_c (mm)	(d) w/o comp. (mm)
Dev.1	1.84	2.74	2.81	2.67
Dev.2	2.01	2.45	2.45	2.30
Dev.3	1.18	1.23	1.30	1.27

(a) decreased, but on the other hand, typically for other methods (b) and (c), the plane fitting errors increased. These results suggest that all parameters considered in our proposed error model are required to improve the quality of the 3D depth data.

Next, we evaluated the method's robustness to errors in the given baseline length w. Our method assumes the target device's baseline length is that given in such articles as Ref. [19]. However, if it is not given, we need to measure it ourselves. In such cases, the measured length may include errors. Thus, we evaluated the robustness to errors in the baseline length w of up to ± 2 mm.

Table 5 shows the errors when the baseline includes errors. As can be seen, our proposed model can reduce errors between the compensated positions and the ground truth positions even when the given baseline length includes errors. This is because our model considers errors in the baseline length w as one of the parameters in Eq. (5).

These experimental results, confirm that our proposed model can improve the quality of 3D depth data obtained by consumer depth cameras such as Kinect and Xtion.

5 Summary

In this paper, we have proposed and evaluated a depth error model for projector-camera based consumer depth cameras such as the Kinect, and

Table 5 Residual errors with varying baseline length errors

Error (mm)	Dev.1 (mm)	Dev.2 (mm)	Dev.3 (mm)
w/o comp.	10.137	5.559	2.144
−2	4.364	3.459	1.659
−1	4.363	3.459	1.665
0	4.361	3.460	1.648
+1	4.359	3.459	1.660
+2	4.357	3.458	1.660

an error compensation method based on calibration of the parameters involved. Since our method only requires depth data without disparity observations, we can apply it to any depth data captured by projector-camera based depth cameras such as the Kinect and Xtion. Our error model considers (i) both camera and projector distortion, and (ii) errors in the parameters used to convert from normalized disparity to depth data. The optimal model parameters can be estimated by showing a chessboard to the depth sensor using multiple arbitrary distances and poses. Experimental results show that the proposed error model can reduce depth measurement errors for both Kinect and Xtion by about 70%. Our proposed model has significant advantages when using a consumer depth camera as a 3D measuring device.

Future work includes further investigation of the error model, improvement of the optimization approach for parameter estimation, and implementation of a calibration tool based on the proposed error model for various projector-camera based depth cameras, such as the Intel RealSense and Occipital Structure Sensor, as well as the Microsoft Kinect.

Appendix A   Derivation of Eq. (10)

Considering errors in the parameters in Eq. (5), $\breve{d}_i$ can be expressed as follows:

$$\breve{d}_i = \breve{f}\breve{w}/\breve{Z}_0 + \breve{m}\delta_i' + \breve{n} \tag{26}$$

where $\breve{f}$, $\breve{w}$, $\breve{Z}_0$, $\breve{m}$, and $\breve{n}$ denote the parameters which contain errors. Let $\Delta(\breve{f}\breve{w}/\breve{Z}_0)$, $\Delta\breve{m}$, and $\Delta\breve{n}$ be the errors in these parameters. Then $\breve{d}_i'$, the ideal value of the observed disparity, can be obtained as follows:

$$\begin{aligned} \breve{d}_i' &= \left(\breve{f}\breve{w}/\breve{Z}_0 + \Delta(\breve{f}\breve{w}/\breve{Z}_0)\right) \\ &\quad + (\breve{m}\Delta\breve{m})\,\delta_i' + (\breve{n} + \Delta\breve{n}) \end{aligned} \tag{27}$$

All parameters except δ_i' are fixed values, so we can obtain Eq. (10) by collecting coefficients:

$$\breve{d}_i' = \alpha\breve{d}_i + \beta \tag{28}$$

Acknowledgements

TSINGHUA UNIVERSITY PRESS Springer

References

[1] Zhang, Z. Microsoft Kinect sensor and its effect. *IEEE Multimedia* Vol. 19, No. 2, 4–10, 2012.

[2] Han, J.; Shao, L.; Xu, D.; Shotton, J. Enhanced computer vision with Microsoft Kinect sensor: A review. *IEEE Transactions on Cybernetics* Vol. 43, No. 5, 1318–1334, 2013.

[3] Smisek, J.; Jancosek, M.; Pajdla, T. 3D with Kinect. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, 1154–1160, 2011.

[4] Herrera, D.; Kannala, J.; Heikkilä, J. Joint depth and color camera calibration with distortion correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 34, No. 10, 2058–2064, 2012.

[5] Yamazoe, H.; Habe, H.; Mitsugami, I.; Yagi, Y. Easy depth sensor calibration. In: Proceedings of the 21st International Conference on Pattern Recognition, 465–468, 2012.

[6] Raposo, C.; Barreto, J. P.; Nunes, U. Fast and accurate calibration of a Kinect sensor. In: Proceedings of the International Conference on 3D Vision, 342–349, 2013.

[7] Xiang, W.; Conly, C.; McMurrough, C. D.; Athitsos, V. A review and quantitative comparison of methods for Kinect calibration. In: Proceedings of the 2nd International Workshop on Sensor-based Activity Recognition and Interaction, Article No. 3, 2015.

[8] Darwish, W.; Tang, S.; Li, W.; Chen, W. A new calibration method for commercial RGB-d sensors. *Sensors* Vol. 17, No. 6, 1204, 2017.

[9] Weiss, A.; Hirshberg, D.; Black, M. J. Home 3D body scans from noisy image and range data. In: Proceedings of the International Conference on Computer Vision, 1951–1958, 2011.

[10] Jin, B.; Lei, H.; Geng, W. Accurate intrinsic calibration of depth camera with cuboids. In: *Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, Vol. 8693.* Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T. Eds. Springer, Cham, 788–803, 2014.

[11] Di Cicco, M.; Iocchi, L.; Grisetti, G. Non-parametric calibration for depth sensors. *Robotics and Autonomous Systems* Vol. 74, 309–317, 2015.

[12] Teichman, A.; Miller, S.; Thrun, S. Unsupervised intrinsic calibration of depth sensors via SLAM. *Robotics: Science and Systems* Vol. 248, 3, 2013.

[13] Wang, H.; Wang, J.; Liang, W. Online reconstruction of indoor scenes from RGB-D streams. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3271–3279, 2016.

[14] Park, J.-H.; Shin, Y.-D.; Bae, J.-H.; Baeg, M.-H. Spatial uncertainty model for visual features using a Kinect^{TM} sensor. *Sensors* Vol. 12, No. 7, 8640–8662, 2012.

[15] Nguyen, C. V.; Izadi, S.; Lovell, D. Modeling Kinect sensor noise for improved 3D reconstruction and tracking. In: Proceedings of the 2nd International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission, 524–530, 2012.

[16] Zhang, Z. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 22, No. 11, 1330–1334, 2000.

[17] Khoshelham, K.; Elberink, S. O. Accuracy and resolution of Kinect depth data for indoor mapping applications. *Sensors* Vol. 12, No. 2, 1437–1454, 2012.

[18] Heikkila, J. Geometric camera calibration using circular control points. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 22, No. 10, 1066–1077, 2000.

[19] Dal Mutto, C.; Zanuttigh, P.; Cortelazzo, G. M. *Time-of-Flight Cameras and Microsoft Kinect^{TM}*. Springer Science & Business Media, 2012.

[20] Freedman, B.; Shpunt, A.; Arieli, Y. Distance-varying illumination and imaging techniques for depth mapping. U.S. Patent 8,761,495. 2014.

Hirotake Yamazoe received his B.E., M.E., and Ph.D. degrees in engineering from Osaka University in 2000, 2002, and 2005, respectively. He was with the Advanced Telecommunications Research Institute International (ATR) during 2005–2011, the Institute of Scientific and Industrial Research, Osaka University during 2011–2012, and Osaka School of International Public Policy, Osaka University during 2012–2015. He is now a lecturer in the College of Information Science and Engineering, Ritsumeikan University. His research interests include computer vision and wearable computing. He is a member of IEICE, IPSJ, ITE, HIS, IEEE, and ACM.

Hitoshi Habe received his B.E. and M.E. degrees in electrical engineering and D.Info. degree in intelligence science and technology from Kyoto University, Japan, in 1997, 1999, and 2006, respectively. After working for Mitsubishi Electric Corporation from 1999 to 2002, Kyoto University from 2002 to 2006, Nara Institute of Science and Technology

from 2006 to 2011, and Osaka University from 2011 to 2012, he is now a lecturer in the Department of Informatics, Kindai University, Japan. From 2010 to 2011, he was a visiting researcher at the Department of Engineering, University of Cambridge, UK. His research interests include computer vision, pattern recognition, and image processing. He is a member of IEEE, ACM, IEICE, IPSJ, and JSAI.

Ikuhisa Mitsugami received his B.S. degree in engineering from Kyoto University in 2001, and M.S. and Ph.D. degrees in engineering from Nara Institute of Science and Technology in 2003 and 2007, respectively. He then started working for the Academic Center for Computing and Media Studies, Kyoto University, and in 2010 became an assistant professor of the Institute of Scientific and Industrial Research, Osaka University. He is currently an associate professor of the Graduate School of Information Sciences, Hiroshima City University. His research interests are computer vision, human interfaces, and human behavior analysis. He is a member of IEEE, IEICE, IPSJ, and VRSJ.

Yasushi Yagi is the Executive Vice President of Osaka University. He received his Ph.D. degree from Osaka University in 1991. In 1985, he joined the Product Development Laboratory, Mitsubishi Electric Corporation, where he worked on robotics and inspection. He became a research associate in 1990, a lecturer in 1993, an associate professor in 1996, and a professor in 2003 at Osaka University. He was also director of the Institute of Scientific and Industrial Research, Osaka University from 2012 to 2015. International conferences for which he has served as Chair include: FG1998 (Financial Chair), OMINVIS2003 (Organizing Chair), ROBIO2006 (Program Co-chair), ACCV2007 (Program Chair), PSVIT2009 (Financial Chair), ICRA2009 (Technical Visit Chair), ACCV2009 (General Chair), ACPR2011 (Program Co-chair), and ACPR2013 (General Chair). He has also served on the IEEE ICRA Conference Editorial Board (2007–2011). He is an editorial board member of IJCV and Editor-in-Chief of *IPSJ Transactions on Computer Vision & Applications.* He was awarded an ACM VRST2003 Honorable Mention Award, IEEE ROBIO2006 Finalist for T.J. Tan Best Paper in Robotics, IEEE ICRA2008 Finalist for Best Vision Paper, MIRU2008 Nagao Award, and PSIVT2010 Best Paper Award. His research interests are computer vision, medical engineering, and robotics. He is a fellow of IPSJ and a member of IEICE, RSJ, and IEEE.

Other papers from this open access journal are available free of charge from http://www.springer.com/journal/41095. To submit a manuscript, please go to https://www.editorialmanager.com/cvmj.