Computer Vision and Image Understanding 139 (2015) 1–20

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Kinect range sensing: Structured-light versus Time-of-Flight Kinect☆

Hamed Sarbolandi*, Damien Lefloch, Andreas Kolb

Institute for Vision and Graphics, University of Siegen, Germany

ARTICLE INFO

Article history:
Received 19 August 2014
Accepted 17 May 2015
Available online 27 May 2015

Keywords:
Depth sensor
3D
Kinect
Evaluation

ABSTRACT

Recently, the new Kinect One has been issued by Microsoft, providing the next generation of real-time range sensing devices based on the Time-of-Flight (ToF) principle. As the first Kinect version was using a structured light approach, one would expect various differences in the characteristics of the range data delivered by both devices.

This paper presents a detailed and in-depth comparison between both devices. In order to conduct the comparison, we propose a framework of seven different experimental setups, which is a generic basis for evaluating range cameras such as Kinect. The experiments have been designed with the goal to capture individual effects of the Kinect devices as isolatedly as possible and in a way, that they can also be adopted, in order to apply them to any other range sensing device. The overall goal of this paper is to provide a solid insight into the pros and cons of either device. Thus, scientists who are interested in using Kinect range sensing cameras in their specific application scenario can directly assess the expected, specific benefits and potential problem of either device.

1. Introduction and related works

In the last decade, several new range sensing devices have been developed and have been made available for application development at affordable costs. In 2010, Microsoft, in cooperation with PrimeSense released a structured-light (SL) based range sensing camera, the so-called Kinect™, that delivers reliable depth images at VGA resolution at 30 Hz, coupled with an RGB-color camera at the same image resolution. Even though the camera was mainly designed for gaming, it achieved great popularity in the scientific community where researchers have developed a huge amount of innovative applications that are related to different fields such as online 3D reconstruction [25,41,43], medical applications and health care [1,15], augmented reality [50], etc. Recently Microsoft released an update of their Kinect™ camera in the context of their next generation of console (XBox One) that is now based on Time-of-Flight (ToF) principle.

Both range sensing principles, SL and ToF, are quite different and are subject to a variety of error sources (see Section 2). This paper is meant to deeply evaluate both Kinect™ cameras, denoted as Kinect^{SL} and Kinect^{ToF} in the following, in order to extract their pros and cons which are relevant for any application incorporating this kind of device. Thus, we explicitly do not try to evaluate the devices with respect to a set of specific application scenarios, but we designed a set of seven different experimental setups as a generic basis for evaluating range cameras such as Kinect.

Several studies can be found in the literature that compare and evaluate the depth precision of both principles. However, this work is the first study comparing both versions of the Kinect cameras and offering detailed descriptions under which conditions one is superior to the other. Since Kinect™ cameras are targeting the consumer market and have known sales of several millions devices, we believe that our work will be valuable for a large number of follow-up research projects.

1.1. Prior work

A complete discussion on prior work in SL- and ToF-based range sensing would clearly go beyond the scope of this paper. Thus, we give a brief and exemplary overview on related work in the context SL- and ToF-based range sensing and focus on papers that compare different range sensing approaches and devices. In Section 2.3 we further refer to some key papers that deal with specific characteristics of SL and ToF range data. Additionally, we refer the reader to the surveys of Berger et al. [3] and Han et al. [17] on the Kinect^{SL} as well as to the survey on Time-of-Flight cameras by Kolb et al. [28].

Kuhnert and Stommel [30] demonstrate a first integration of ToF- and stereo cameras. Beder et al. [2] evaluate and compare ToF

☆ This paper has been recommended for acceptance by Pushmeet Kohli.

* Corresponding author.
E-mail address: hamed.sarbolandi@uni-siegen.de, hamed.sarbolandi@gmail.com (H. Sarbolandi).

http://dx.doi.org/10.1016/j.cviu.2015.05.006
1077-3142/

cameras to a stereo-vision setup. Both papers emphasize that ToF and stereo data are at least partially complementary and thus an integration significantly improves the quality of range data. Furthermore, the Kinect^{ToF} does not use triangulation for depth calculation, and thus it does not suffer much from occlusion. As it will be shown in Section 4.8, the occluded area in a static scene is around 5% compared to Kinect^{SL} which is around 20%. Besides Evangelidis et al. [11] have also used a ToF range camera, in comparison with Kinect^{SL}, Kinect^{ToF} would be a better choice specifically to be utilized in depth-stereo approach. For further details on ToF-stereo fusion we refer the reader to Nair et al. [40]. In the domain of robotics, Wiedemann et al. [51] compare different ToF cameras from different manufacturers. They analyze the sensor characteristics of such systems and the application potential for mobile robots. In their work, they address several problems such as sensor calibration, automatic integration time and data filtering schemes for outliers measurements removal. Stoyanov et al. [48,49] compare the accuracy of two ToF cameras and the Kinect^{SL} camera to a precise laser range sensor (aLRF). However their evaluation methodology does not take into account the different error sources given by real-time range sensing cameras. The follow-up work by Langmann et al. [31] compares a ToF camera (pmdtec CamCube 41k) with the Kinect^{SL}. Lateral resolution of depth measurements are given using a three dimensional Siemens star-like shape. The depth linearity is also compared using precise linear rail. The authors conclude that both cameras have different drawbacks and advantages and thus are meant to be used for different applications. Meister et al. [38] discuss the properties of the 3D data acquired with a Kinect^{SL} camera and fused into a consistent 3D Model using the so-called KinectFusion-pipeline [41] in order to provide ground truth data for low-level image processing. The "targetbox" scene used by Meister et al. [38], also called "HCI Box", consists of several object arranged in a $1 \times 1 \times 0.5$ m box. Nair et al. [39] discuss quality measures for good ground truth data as well as measurement and simulation approaches to generate this kind of data. We generally opted against this kind of ground truth scenery, as this approach does often not allow a proper separation of the individual error sources and, thus, it would be nearly impossible to transfer results to another application scenario.

In their book about ToF cameras Hansard et al. [18] compare between ToF cameras and the Kinect^{SL}. Their comparison focuses on different material classes. They use 13 diffuse ("class A"), 11 specular ("class B") and 12 translucent ("class C") objects or object variants for which they acquire geometric ground truth using an additional 3D scanner and applying white matte spray on each object surface. As result, they provide root mean square error (RMSE) and standard deviation (SD).

Compared to all prior work, in this paper we focus on a set of experimental setups handling an as complete as possible list of characteristic sensor effects and evaluate these effects for the Kinect^{SL} and the Kinect^{ToF} cameras presented in Section 2. Before presenting the experiments and results, we discuss the fundamental problem raised by any attempt to compare these devices in Section 3. In Section 4 we present our experiments, that are all designed in such a way that individual sensor effects can be captured as isolatedly as possible and that the experiments are reproducible for other range sensing cameras.

2. Devices principle

2.1. Structured light cameras: Kinect^{SL}

Even though the principle of structured light (SL) range sensing is comparatively old, the launch of the Microsoft Kinect^{TM} (Kinect^{SL}) in 2010 as interaction device for the XBox 360 clearly demonstrates the maturity of the underlying principle.

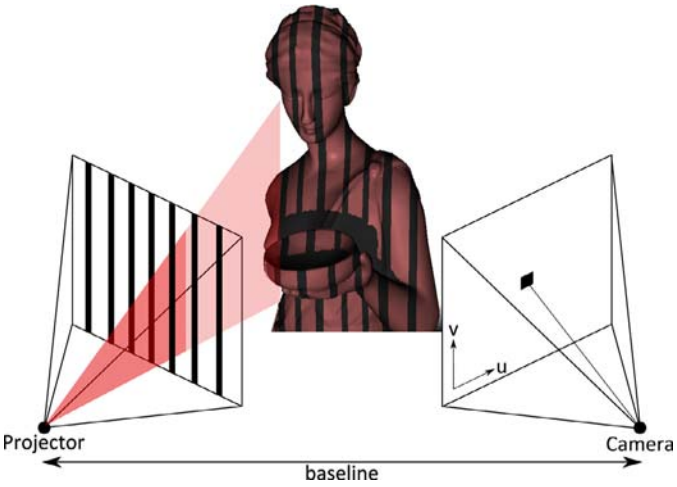

Fig. 1. Principle of structured light based systems.

2.1.1. Technical foundations

The structured light approach is an active stereo-vision technique. A sequence of known patterns is sequentially projected onto an object, which gets deformed by geometric shape of the object. The object is then observed from a camera from a different direction. By analyzing the distortion of the observed pattern, i.e. the disparity from the original projected pattern, depth information can be extracted; see Fig. 1.

Knowing the intrinsic parameters of the camera, i.e. the *focal length f* and additionally the *baseline b* between the observing camera and the projector, the depth of pixel (x, y) can be computed using the disparity value $m(x, y)$ for this pixel as $d = \frac{b \cdot f}{m(x,y)}$. As the disparity $m(x, y)$ is usually given in pixel-units, the focal length is also converted to pixel units, i.e. $f = \frac{f_x^{\text{metric}}}{s_{\text{px}}}$, where s_{px} denotes the pixel size. In most cases, the camera and the projector are only horizontally displaced, thus the disparity values are all given as horizontal distances. In this case s_{px} resembles the horizontal pixel size. The depth range and the depth accuracy relate to the baseline, i.e. longer baselines allow for robust depth measurements at long distances.

There are different options to design the projection patterns for a SL range sensor. Several approaches were proposed based on the SL principle in order to estimate the disparity resulting from the deformation of the projected light patterns. In the simplest case the stripe-pattern sequence realizes a binary code which is used to decode the direction from an object point is illuminated by the beamer. Based on this principle, Hall-Holt and Rusinkiewicz [16] introduced a real-time camera based 3D system. The authors show that they could achieve full 3D reconstruction of objects using an automatic registration of different rotated range maps.

Zhang et al. [55] investigate the benefit of projection patterns composed of alternative color stripes creating color transitions that are matched with observed edges. Their matching algorithm is faster and eliminates the global smoothness assumptions from the standard SL matching algorithm. Similarly, Fechteler et al. [13] use this color pattern to reconstruct at high-resolution human face using only two sequential patterns, which leads to a reduced computational complexity.

Additionally, Zhang and Huang [56] propose a high resolution SL camera based on the use of color fringes pattern and phase-shifting techniques. Their system was designed to capture and reconstruct at high frame rate (up to 40 Hz) dynamic deformable objects such as human face.

SL cameras, such as the Kinect^{SL}, use a low number of patterns, maybe only one, to obtain a depth estimation of the scenery at a "high" frame rate (30 FPS). Typically, it is composed of an near

Fig. 2. Sensor placement within a Kinect^{SL} camera. The baseline is of approximately 7.5 cm.

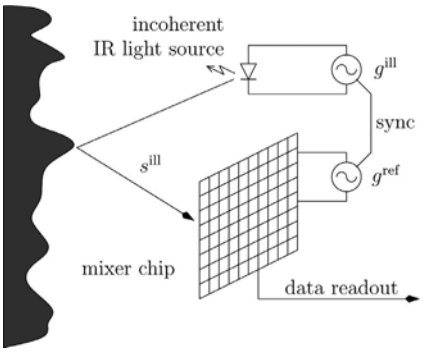

Fig. 3. The ToF phase-measurement principle.

infra-red (NIR) laser projector combined with a monochrome CMOS camera which captures depth variations of object surfaces in the scene.

The Kinect^{SL} camera is based on the standard structured light principle where the device is composed of two cameras, i.e. a color RGB and a monochrome NIR camera, and an NIR projector including a laser diode at 850 nm wavelength. The baseline between the NIR projector and the NIR camera is 7.5 cm, see Fig. 2. The NIR projector uses a known and fixed dot pattern to illuminate the scenery.

Simple triangulation techniques are later on used to compute the depth information between the projected pattern seen by the NIR camera and the input pattern stored on the unit. For each pixel p_i, depth is estimated by finding the best correlation pattern patch, typically in a 9×9 pixel window, on the NIR image with the corresponding projection pattern. The disparity value is given by this best match. Note that the Kinect^{SL} device performs internally an interpolation of the best match operation in order to achieve sub-pixel accuracy of $\frac{1}{8}$ pixel. A detailed description of the Kinect disparity map computation can be found at the ROS.org community website [29], where the Kinect^{SL}'s disparity map computation has been reverse engineered and a complete calibration procedure is deduced.

2.2. Time-of-flight (ToF) cameras

The ToF technology is based on measuring the time that light emitted by an illumination unit requires to travel to an object and back to the sensor array [32]. In the last decade, this principle has found realization in microelectronic devices, i.e. chips, resulting in new range-sensing devices, the so-called *ToF cameras*. Here, we will explain the basic principle of operation of ToF-cameras. It should be noted that for the specific device of the new Kinect^{ToF} camera, issued by Microsoft Corp. in conjunction with the XBox 360 game console, only little technical detail is known.

The Kinect^{ToF} utilizes the *Continuous Wave (CW) Intensity Modulation* approach, which is most commonly used in ToF cameras. The general idea is to actively illuminate the scene under observation using near infrared (NIR) intensity-modulated, periodic light (see Fig. 3). Due to the distance between the camera and the object (sensor and illumination are assumed to be at the same location), and the finite speed of light c, a *time shift* $\phi[s]$ is caused in the optical signal which is equivalent to a *phase shift* in the periodic signal. This shift is detected in each sensor pixel by a so-called *mixing* process. The time shift can be easily transformed into the sensor-object distance as the light has to travel the distance twice, i.e. $d = \frac{c\phi}{4\pi}$.

From the technical perspective, the generator signal g^{ill} driving the illumination unit results in the intensity modulated signal which, after being reflected by the scene, results in an incident optical signal s^{ill} on each sensor pixel. Note that the optical signal may be deformed by nonlinear effects e.g. in the LEDs of the illumination unit. The incident signal s^{ill} is correlated with the reference generator signal g^{ref}. This mixing approach yields the correlation function which is sampled in each pixel

$$C[g^{ill}, g^{ref}] = s \otimes g = \lim_{T\to\infty} \int_{-T/2}^{T/2} s^{ill}(t) \cdot g^{ref}(t)\, \mathrm{d}t.$$

The phase shift is computed using several correlation measurements with varying illumination and reference signals g_i^{ill} and g_i^{ref}, respectively, using some kind of demodulation function, i.e.

$$\phi = \mathcal{G}(A_0, A_1, \ldots, A_n), \quad \text{with } A_i = C[g_i^{ill}, g_i^{ref}], i = 1, \ldots, n.$$

Frequently, A_i is called *phase image* or *correlation image*. We will use the latter notation in order to prevent confusion with the phase shift ($\propto$ distance). Practically, the correlation images are acquired sequentially, however there is the theoretic option to acquire all correlation images in parallel, e.g. by having different phase shifts for neighboring pixels. Note that due to the periodicity of the reference signal, any ToF-camera has a unique unambiguous measurement range.

The first ToF cameras like the prototypes from pmdtechnologies [53] used sinusoidal signals $g^{ill}(t) = \cos(2\pi f_m t)$ with a constant modulation frequency f_m and a reference signal equal to g^{ill} with an additional phase offset τ, i.e. $g^{ref}(t) = g^{ill}(t + \tau)$. For this approach, usually four correlation images $A_i = C[\cos(2\pi f_m \cdot), \cos(2\pi f_m \cdot + \tau_i)]$ for $\tau_i = i \cdot \pi/2,\ i = 0, 1, 2, 3$ are acquired leading to a distance value of

$$\phi = \mathcal{G}(A_0, A_1, A_2, A_3) = \operatorname{arctan2}(A_3 - A_1, A_0 - A_2)/f_m,$$

where $\operatorname{arctan2}(y, x)$ is the angle between the positive x-axis and the point given by the coordinates (x, y).

The Kinect^{ToF} camera applies this CW intensity modulation approach [52]. Blake et al. [5] reverse engineered the Kinect^{ToF}-driver. This revealed that the Kinect^{ToF} acquires 10 correlation images, from which nine correlation images are used for a three-phase reconstruction approach based on phase shifts of 0°, 120° and 240° at three different frequencies. Using multiple modulation frequencies the measurement range can be exceeded [9] Although the Kinect^{ToF} camera can obtain depth values for distances longer than 9 m, the official driver masks the distances further than around 4.5 m.

The purpose of the tenth correlation image is still not clear. Even though the technical specifics of the Kinect^{ToF} have not been explicitly revealed by Microsoft, it definitely applies the basic principle of correlation as described above. The illumination unit consists of a laser diode at 850 nm wavelength.

In Section 3 we discuss further technical details regarding the Kinect^{ToF} driver.

2.3. Error sources for Kinect^{SL} and Kinect^{ToF}

SL and ToF cameras are active imaging systems that use standard optics to focus the reflected light onto the chip area. Therefore, the typical optical effects like shifted optical centers and lateral distortion need to be corrected, which can be done using classical intrinsic camera calibration techniques. Beyond this camera specific calibration issues, SL and ToF cameras possess several specific error sources, which are discussed in the following and which also apply to Kinect^{ToF} and/or Kinect^{SL}. As a detailed discussion of prior work in

relation to these error sources would go beyond the scope of this paper, we only give some relevant links to prior work that relates to the individual effects for either system.

2.3.1. Ambient background light [SL, ToF]

As any other camera, ToF and SL cameras can suffer from ambient background light, as it can either lead to over-saturation in case of too long exposure times in relation to the objects' distance and/or reflectivity, e.g. causing problems to SL-systems in detecting the light pattern. Both, the Kinect^{ToF} and the Kinect^{SL} are utilized with a band-pass filter, suppressing background light out of the range of the illumination. Kinect^{ToF} provides a suppression of background intensity on the chip.

For ToF cameras specific circuitry has been developed, e.g. the *Suppression of Background Intensity* approach for PMD cameras [44] that electronically filter out the DC-part of the light. For SL systems outdoor application is usually hard to achieve, which has also been stated for the Kinect^{SL} [10].

2.3.2. Multi-device interference [SL, ToF]

Similar to any other active sensing approach, the parallel use of several Kinect cameras may lead to interference problems, i.e. the active illumination of one camera influences the result of another camera.

For Kinect^{SL} the potential interference problem given by multiple NIR patterns projected into the scene is very difficult to solve. Butler et al. [7] propose a "Shake'n'Sense" setup where one (or each) Kinect^{SL}-device is continuously shaken using an imbalanced rotating motor. Thus, the projected pattern performs a high frequency motion that appears significantly blurred for another device. An alternative approach is introduced by Berger et al. [4]. They add steerable hardware shutters to the Kinect^{SL}-devices' illumination units resulting in a time-multiplex approach. For ToF cameras the signal shape can be altered in order to prevent multi-device interference, e.g. for sinusoidal signal shapes different modulation frequencies can simply be used to decouple the devices [27].

2.3.3. Temperature drift [SL, ToF]

A common effect to many technical devices is the drift of the system output, i.e. the distance values in the case of Kinect cameras, during the device warm-up. The major difference between the SL and the ToF approach is that an SL camera usually does not produce as much heat as a ToF camera. This is due to the fact that the required illumination power to cover the full scene width and depth in order to get a sufficient *signal-to-noise* (SNR) for the optical signal for a ToF camera is beyond the power needed to generate the relatively sparse point-based pattern applied by the Kinect^{SL}. As a consequence, the Kinect^{SL} can be cooled passively whereas the Kinect^{ToF} requires active cooling.

For the Kinect^{SL} significant temperature drift has been reported by Fiedler and Müller [14]. Early ToF-camera studies e.g. from Kahlmann et al. [24] of the Swissranger^{TM} camera exhibit the clear impact of this warm-up on the range measurement. More recently smaller ToF cameras for close range applications such as the camboard-nano series provided my pmdtechnolgies do not require active cooling, however, no temperature drift investigations have been reported so far.

2.3.4. Systematic distance error [SL, ToF]:

Both Kinect cameras suffer from systematic error in their depth measurement. For the Kinect^{SL} the error is mainly due to inadequate calibration and restricted pixel resolution for estimation of the point locations in the image plane, leading to imprecise pixel coordinates of the reflected points of the light pattern [26]. Further range deviations for the Kinect^{SL} result from the comparably coarse quantization of the depth values which increase for further distances from the camera. For Kinect^{ToF}, on the other hand, the distance calculation based on the mixing of different optical signals s with reference signals g^{ref} requires either an approximation to the assumed, e.g. a sinusoidal signal shape or an approximation to the phase demodulation function $\mathcal{G}$. Both approximations lead to a systematic error in the depth measurement. In case of an approximated sinusoidal shape this effect is also called "wiggling" (see Fig. 4, top left). The systematic error may depend on other factors, such as the exposure time.

For Kinect^{SL} Khoshelham and Elberink [26] present a detailed analysis of its accuracy and depth resolution. They conclude that the systematic error is below some 3 cm, however it increases on the periphery of the range image and for increasing object-camera distance. Smisek et al. [47] present a geometric method to calibrate the systematic error of the Kinect^{SL}. Herrera et al. [19] proposed a joint calibration approach for the color and the depth camera of the Kinect^{SL}. Correction schemes applied to reduce the systematic error of ToF cameras with sinusoidal reference signals simply model the depth deviation using a look-up-table [23] or function fitting, e.g. using b-splines [33].

2.3.5. Depth inhomogeneity [SL, ToF]

At object boundaries, a pixel may observe inhomogeneous depth values. Due to the structured light principle, occlusion may happen at object boundaries where parts of the scene are not illuminated by the infra-red beam which results in a lack of depth information in those regions (invalid pixels). For ToF cameras, the mixing process results in a superimposed signal caused by light reflected from different depths, so-called *mixed pixels*. In the context of ToF cameras these pixels are sometimes called *flying pixels*. The mixed or flying signal leads to wrong distance values; see Fig. 4, top right.

There are simple methods relying on geometric models that give good results in identifying flying pixel, e.g. by estimating the depth variance which is extremely high for flying pixel [45]. Denoising techniques, such as a median filter, can be used to correct some of the flying pixels.

Note that flying pixels are directly related to a more general problem, i.e. the multi-path problem; see below.

2.3.6. Multi-path effects [SL, ToF]

Multi-path effects relate to an error source common to active measurement systems: The active light may not only travel the direct path from the illumination unit via the object's surface to the detector, but may additionally travel *indirect paths*, i.e. being scattered by highly reflective objects in the scene or within the lens systems or the housing of the camera itself, see Fig. 4 bottom left. In the context of computer graphics this effect is known as global illumination. For ToF cameras these multiple responses of the active light are superimposed in each pixel leading to an altered signal not resembling the directly reflected signal and thus a wrong distance. For Kinect^{SL} indirect illumination mainly causes problems for highly reflecting surfaces, as dots of the pattern may be projected at other objects in the scene. However, objects with a flat angle to the camera will lead to a complete lack of depth information (see also Section 4.7).

For ToF cameras several correction schemes for multi-path effects have been proposed for sinusoidal signal shapes. Falie and Buzuloiu [12] assume that the indirect effects are of rather low spatial frequency and analyze the pixel's neighborhood to detect the low-frequency indirect component. Dorrington et al. [8] present an analytic formulation for the signal superposition resulting in a non-linear optimization scheme per pixel using different modulation frequencies.

2.3.7. Intensity-related distance error [ToF]

Considering a highly reflecting object and a second object with the same distance to the camera but with low reflectivity in the relevant NIR range, a reduced SNR is expected. Beyond this, it has frequently been reported that ToF cameras have a non-zero biased distance offset for objects with low NIR reflectivity (see Fig. 4, bottom right).

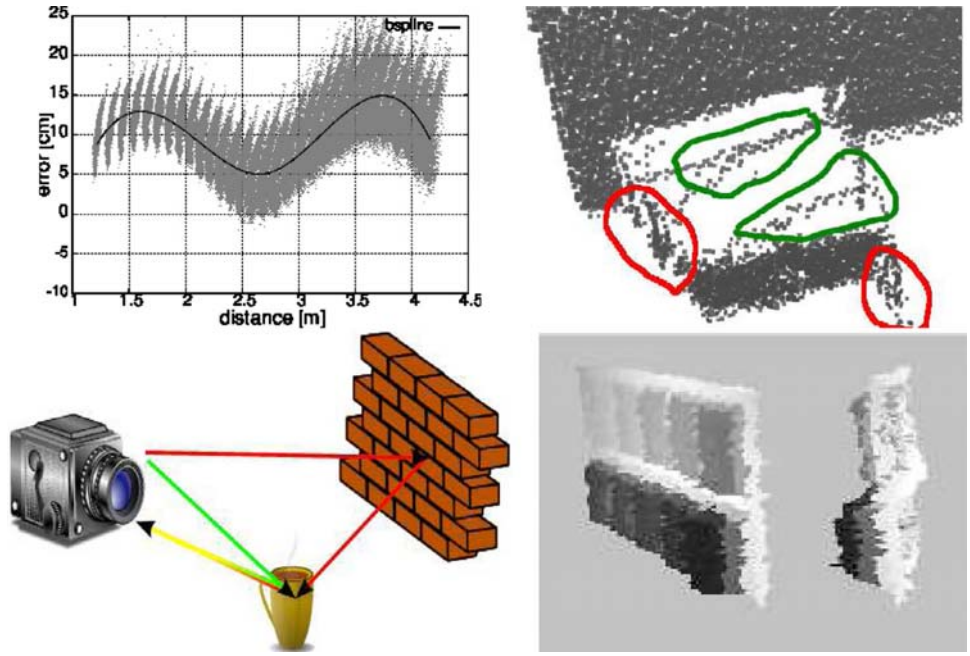

Fig. 4. Error sources of ToF cameras. Top left: Systematic (wiggling) error for all pixels (gray) and fitted mean deviation (black). Top right: Motion artifacts (red) and flying pixels (green) for a horizontally moving planar object in front of a wall. Bottom left: Schematic illustration of multi-path effects due to reflections in the scene. Bottom right: Acquisition of a planar gray-scale checkerboard reveals the intensity related distance error. (Image courtesy: [28], Eurographics Association, 2010.) (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Lindner et al. [35] tackle the specific intensity-related error using phenomenological approaches. In general, there are at least two possible explanations for this intensity-related effect. The first assumption explains this effect is a specific variant of a multi-path effect, the second one puts this effect down to the non-linear pixel response for low amounts of incident intensity.

2.3.8. Semitransparent and scattering media [SL, ToF]

As for most active measuring devices, media that does not perfectly reflect the incident light potentially causes errors for ToF and SL cameras. In case of ToF cameras, light scattered within semitransparent media usually leads to an additional phase delay due to a reduced speed of light.

The investigations done by Hansard et al. [18] give a nice overview for specular and translucent, i.e. semitransparent and scattering media for ToF cameras with sinusoidal reference signal and the Kinect^{SL}. Kadambi et al. [22] show that their coding method (originally designed to solve multi-path errors for ToF cameras) is able to recover depth of near-transparent objects using their resulting time-profile (transient imaging). Finally, a detailed state-of-the-art report is given by Ihrke et al. [21] where different methods are described in order to robustly acquire and reconstruct such challenging media.

2.3.9. Dynamic scenery [SL, ToF]

One key assumption for any camera-based system is that each pixel observes a single object point during the whole acquisition process. This assumption is violated in case of moving objects or moving cameras, resulting in motion artifacts. In real scenes, motion may alter the true depth. Even though Kinect^{SL} acquires depth using only a single NIR image of the projected pattern, a moving object and/or camera leads to improper detection of the pattern in the affected region. ToF cameras as the Kinect^{ToF} require several correlation images per depth image. Furthermore, their correlation measurements get affected by a change of reflectivity observed by a pixel during the acquisition. Processing the acquired correlation images ignoring the motion present during acquisition leads to erroneous distance values at object boundaries (see Fig. 4, top right).

However, no real investigations have been done yet for the Kinect^{SL} to study the effect of motion blur on the depth measurement quality. Nevertheless the work of Butler et al. [7] uses the motion blur property to solve the problem of multiple Kinect^{SL} devices interference.

For ToF cameras several motion compensation schemes have been proposed. Schmidt and Jahne [46] detect motion artifacts using temporal gradients of the correlation images A_i, i.e. a large gradient in one of the correlation images indicates motion. This approach also performs a correction using extrapolated information from prior frames; see also discussion in Hansard et al. [18], Section 1.3.3. Since motion artifacts result from in-plane motion between subsequent correction images, several approaches use optical flow methods in order to realign the individual correlation images. Lindner and Kolb [34] apply a fast optical flow algorithm [54] three times in order to align the four correlation images A_0, A_1, A_2, A_3 to the first correlation image A_0. As optical flow algorithms are computationally very expensive, these approaches significantly reduce the frame rates for real-time processing. A faster approach is motion detection and correction using block-matching techniques applied pixels where motion has been detected [20].

3. General considerations for comparing Kinect^{SL} and Kinect^{ToF}

Before presenting the experimental setups and the comparison between the two Kinect devices, we have to consider the limitations which this kind of comparison encounters. For both, the Kinect^{SL} and the Kinect^{ToF} cameras, there are no official, publicly available reference implementations which explain all stages from raw data acquisition to the final range data delivery. Thus, any effect observed may either relate to the sensor hardware, i.e. to the measurement principle as such, or to the algorithms applied to raw data or, in a post-processing manner, to the range data which is integrated in the camera systems.

Anticipating the further discussion in this section, we explicitly opted to work with both Kinect cameras in a “black box” manner using the official drivers, as it is impossible to achieve “fair conditions”

for the comparison, i.e. a comparison which neutralizes the effects from diverse filters applied in range data processing. This is mainly due to the fact that data processing is applied on the level of raw data, i.e. disparity maps or correction images, as well as on the level of range data; see detailed discussion below. Attempts to reverse engineer the processing functionality usually do not lead to the same data quality; see below. Thus, taking the devices as they are, including the official, closed-source drivers, is the most appropriate approach from the perspective in utilizing them for any kind of application.

However, the disparity map from the Kinect^{SL} is different from common representation, i.e. 0 disparity value does not refer to an infinite distance. According to the reverse engineered disparity map

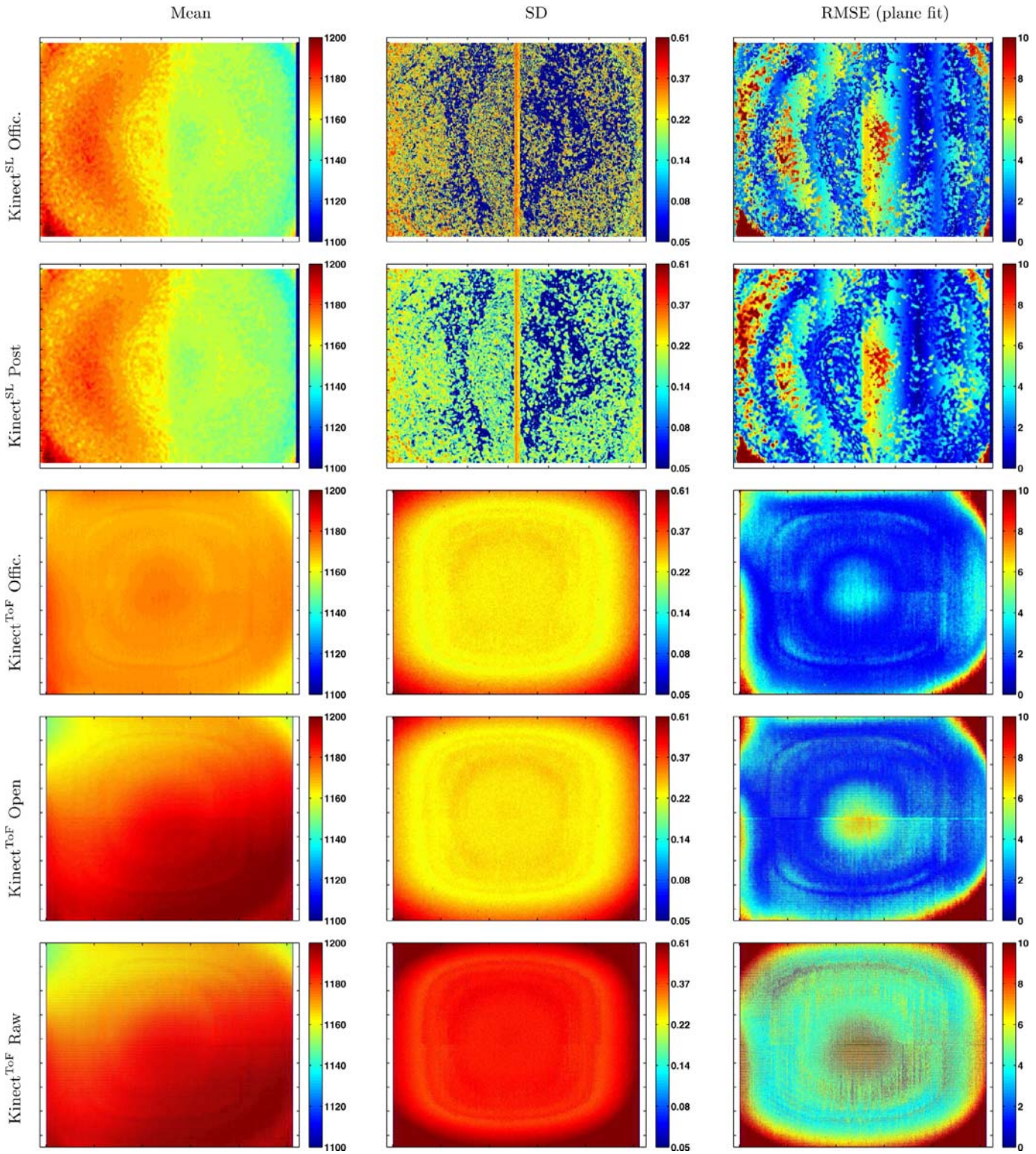

Fig. 5. Statistics of 200 frame for Kinect^{SL} and Kinect^{ToF} acquiring a planar wall (values in mm): Mean (left col.), standard deviation (middle col.) and RMSE with respect of a fitted plane (right) for the Kinect^{SL} (official driver, 1st row, post-filtered range images, 2nd row) and for the Kinect^{ToF} (official driver, 3rd row, the re-engineered OpenKinect driver, 4th row, and the raw range data delivered by the OpenKinect driver, 5th row).

computation from ROS.org, the disparity map is normalized and quantized between 0 and 2047 (using 11 bits storage), that requires a more complex mapping function in order to convert disparity into depth values. Note that the quantization of the disparity map leads to quantization of range values, which in some cases negatively influences the statistical analysis or, in some cases, makes it completely useless. For example, it is impossible to derive a per-pixel noise model for the Kinect^{SL} taking only individual pixel distance measurements of a static scene; see Section 4.1 and Nguyen et al. [42].

Different alternatives have been proposed for depth value estimation for Kinect^{SL} disparity maps [26]. In general, it is possible to access the raw data of the Kinect^{SL} camera, i.e. infrared image of the scene with the dots pattern, but it would go far beyond the scope of this paper to provide further insight into the Kinect^{SL}'s method of operation by reverse engineering. On the other hand, solely post-processing the delivered range data hardly improves the quality; see below.

As described in Section 2.2. the Kinect^{ToF} camera applies the CW approach. Additionally, the reverse engineered OpenKinect driver [5] gives insight into some details of data processing applied in the Kinect^{ToF}. In a first processing stage, the correction images are converted to intermediate images. At this stage a bilateral filter is applied. In a second stage, the final range data is computed out of the intermediate images, joining the three different range values retrieved from the three frequencies. At this level, an outlier (or flying pixel) removal is applied. The OpenKinect driver allows to deactivate the two filters, thus the delivered range data can be considered as being based raw correction images.

The described functionality allows data access on several levels, i.e.

- *Kinect^{SL} Offic* and *Kinect^{ToF} Offic*: Range data as delivered by the official driver provided by Microsoft for the Kinect^{SL}[1] and for the Kinect^{ToF} using the Developer Preview driver.[2]
- *Kinect^{SL} Post*: Additional post-processing using filtering; here we use a bilateral filter. Note that the filter has to operate on data with already masked out, i.e. invalid pixels.
- *Kinect^{ToF} Open*: The reengineered data processing of the OpenKinect driver by Blake et al. [5].
- *Kinect^{ToF} Raw*: The reengineered data processing of the OpenKinect driver by Blake et al. [5] with deactivated filtering, i.e. range data directly computed from the raw data.

We apply these five different options to a simple static scene, where the cameras observe a planar wall, analyzing the statistics for 200 frames; see Fig. 5 and Section 4.8). For this scenario, the data is comparable among different drivers of a device, as the cameras have not been moved while switching to a different driver. However, the data is not fully comparable between Kinect^{SL} and Kinect^{ToF}. Additionally, we used a dynamic scenery with a rotating Siemens star; see Fig. 6 and Section 4.8).

The results for the static wall and the Siemens star are presented in Figs. 5 and 6, respectively. The results can be summarized as follows:

- Post-processing the Kinect^{SL}-data does not improve the quality, as the problematic regions of the range image are already masked out; see Fig. 6, top row. The quality of the Kinect^{SL} device is mainly driven by strong depth quantization artifacts, which get apparent in the standard deviation; see Fig. 5, middle column, first two rows.
- The quality of the OpenKinect driver [5] stays somewhat behind the official Kinect^{ToF}-driver; see Fig. 5, 3rd and 4th rows, i.e. the reverse engineering appears to be functionally not fully complete.

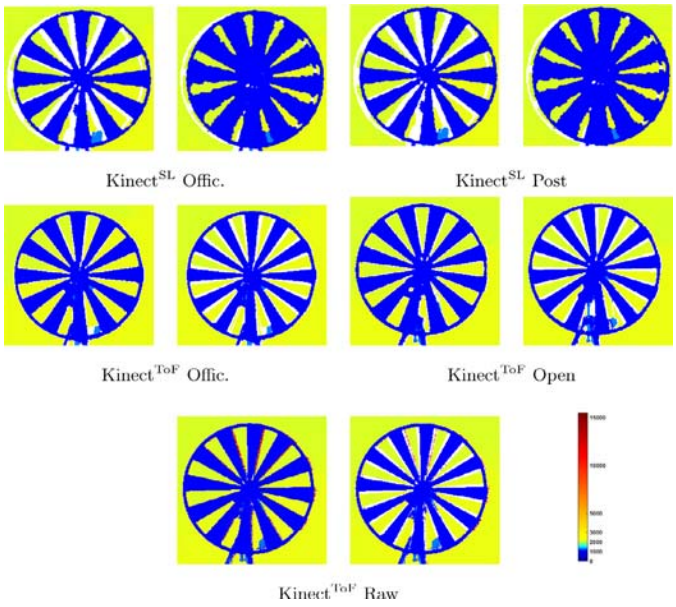

Fig. 6. Single depth frame for a Siemens star for Kinect^{SL} and Kinect^{ToF}, range in mm: The range images are acquired for the static (left image) and the rotating star (60 RPM, right image) for Kinect^{SL} (official driver, top left, and post-filtered range images, top right) and for the Kinect^{ToF} (official driver, middle left, the re-engineered OpenKinect driver, middle right, and the raw range data delivered by the OpenKinect driver, bottom). White color indicates invalid pixels.

- Disabling the internal filters for the Kinect^{ToF} mainly shows negative effects for the rotating Siemens star; see Fig. 6. The filtering of the correction images and the flying pixel removal clearly removes the artifacts at the jumping edges of the Siemens star.

4. Experimental results and comparison

In Sections 4.2–4.8 we present the different test scenarios we designed in order to capture specific error sources of the Kinect^{SL} and the Kinect^{ToF}-cameras. Before going into the scenarios, in Section 4.1 we will briefly present the camera parameters and the pixel statistics.

Our major design goal for the test scenarios was to capture individual effects as isolatedly as possible. Furthermore, we designed the scenarios in a way that they can be reproduced in order to adopt them to any other range sensing system that works in a similar depth range. Table 1 gives an overview of the different test scenarios and the effects they address; see also Section 2.3. We focus on the range of 500 mm–3000 mm[3] as we operate the Kinect^{SL} in the so-called near-range-mode, which is optimized for this depth range. Also it covers the depth range supported by Kinect^{ToF} which is 500 mm–4500 mm.[4]

For all tests we utilize a Kinect^{SL} (Kinect for Windows v1 sensor with activated near mode) and a Kinect^{ToF} (Microsoft camera prototype available from the Developer Preview Program). Data access for the Kinect^{SL} is done via the official driver provided by Microsoft[5] and for the Kinect^{ToF} using the Developer Preview driver.[6] All data evaluations have been done using Matlab.

The major quantitative results for the comprehensive comparison are summarized in Table 4 indicating the major differences, strengths and limitations of both systems.

At this point, we want to refer to the discussion in Section 3 state explicitly, that both Kinect cameras are used in a "black box" manner. Thus, even though we refer to characteristics of the specific range

[1] Kinect for Windows SDK 1.8.
[2] Kinect for Windows SDK 2.0 (JuneSDK).
[3] Microsoft Developer Network, Kinect sensor.
[4] Kinect for Windows, features.
[5] Kinect for Windows SDK 1.8.
[6] Kinect for Windows SDK 2.0 (JuneSDK).

Table 1
The different effects relevant SL- and ToF-based range sensing systems and their relation to the designed test scenarios. Each test addresses primarily one or two separable effects denoted by • and may address also secondary effects, denoted by ○.

Test-scenarios\effect	Amb. backgr. light	multi-device interf.	Temperature drift	Systematic error	Depth inhomogeneity	Multipath effect	Intens.-rel. error	Semitrans. and scatter.	Dynamic scenery
Ambient background light		•							
Multi-device interference	•	○							
Device warm-up			•						
Rail depth tracking				•			•		
Semitransparent liquid						○		•	
Reflective board						•			
Turning Siemens star					•				•

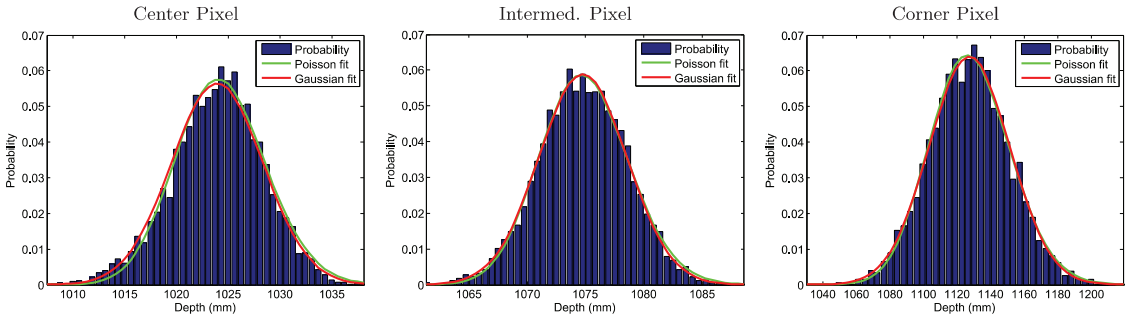

Fig. 7. Density distribution of 3 different pixels during 5000 frames for Kinect^{ToF} acquiring a planar wall (values in mm).

Table 2
Camera parameters of the depth sensors for Kinect^{SL} and Kinect^{ToF}. The distortion coefficients are radial (k_1, k_2, k_3) and tangential distortion (p_1, p_2).

Parameter	Kinect^{SL}	Kinect^{ToF}
Resolution	640 × 480	512 × 424
Focal length (px)	(583.46, 583.46)	(370.79, 370.20)
Center (px)	(318.58, 251.55)	(263.35, 202.61)
Dist. (k_1; k_2; k_3; p_1; p_2)	−0.07377, 0.1641, 0, 0, 0	0.09497, −0.2426, 0, 0.00076, −0.00017

Table 3
Temporal statistics of three different pixels of Kinect^{ToF} sensor with the corresponding Gaussian and Poisson fits. The value δ_x (mm) denotes the shift applied range values to match the Poisson distribution.

Pixel	Gaussian (μ, σ)	RMSD$_g$	Poisson (λ, δ_x)	RMSD$_p$
Center	[1023.91, 4.42]	0.0025	[17.47, 1007.47]	0.0024
Intermed.	[1074.77, 3.77]	0.0017	[14.10, 1061.40]	0.0017
Corner	[1127.21, 24.03]	0.0019	[101.73, 1030.02]	0.0019

measurement techniques, the resulting effects may not only relate to the sensor hardware, i.e. to the measurement principle as such, but also to the post-processing integrated into the cameras.

4.1. Camera parameters and noise statistics

As most applications require full 3D information, we first estimate the intrinsic parameters for both devices using standard calibration techniques based on a planar checkerboard from the OpenCV library [6]; see Table 2. For both devices, 50 images of the checkerboard were acquired with different orientations and distances. For the Kinect^{ToF}, we directly use the amplitude image delivered by the camera. Whereas for the Kinect^{SL}, we use the NIR image of the depth sensor. Since the dot pattern of the Kinect^{SL} may degrade the checkerboard detection quality in the NIR image, we block the laser illumination and illuminate the checkerboard with an ambient illumination.

Furthermore, we want to analyze the noise statistics for the Kinect^{SL} and the Kinect^{ToF}. As already stated in Section 3, the strong quantization applied in the Kinect^{SL} makes it hard to derive per-pixel temporal statistic values. In the literature there are alternative approaches using a mixed spatiotemporal analysis to derive some kind of noise statistics [42], but this approach is difficult to compare with pure temporal statistics. Therefore, we focus on the Kinect^{ToF}'s temporal statistics only.

For the temporal statistics we acquired 5000 frames of the Kinect^{ToF} observing a planar wall at about 1 m distance. The OpenKinect driver with deactivated filtering was used to obtain unchanged range data. Fig. 7 shows the histograms for a central, an intermediate and a corner pixel of this time series including fits for a Gaussian and a Poisson distribution. Both fits were done using Matlab. We use non-linear least square optimization approaches in order to get the suitable parameters for the Poisson distribution. Table 3 gives the resulting parameters of both fitting for the three pixel statistics as well as the corresponding RMSE. It can be noted that corner pixels have a higher variance than pixel at the center area of the image, which is due to a reduced amplitude of the illumination in corner regions. We can also deduce that the Poisson distribution and the Gaussian fitting results in the same fitting quality.

4.2. Ambient background light

4.2.1. Goal

This test scenario addresses the influence of ambient light onto the range measurement of the Kinect^{SL} and the Kinect^{ToF} cameras. The primary goal for the experiment is to show the relation between ambient background radiance incident to the Kinect and the delivered depth range of the sensor. The main focus for this experiment is thus to measure the incident background radiance with respect to image regions accurately.

As both Kinect cameras do have imperfect illumination in the sense, that pixels in the vicinity on the image receive less active light than pixels close to the center, a secondary goal is to give some insight into a possible spatial variation of the influence of ambient background light.

4.2.2. Experimental setup

The Kinect camera is mounted 1200 mm in front of an approximately diffuse white wall in an environment where the amount of light can be controlled using three HALOLINE ECO OSRAM 400 W

Table 4
Summarizing the major Kinect characteristics. IP="invalid pixel", SD="standard deviation", RMSE="root mean square error", SDA="standard deviation average", SE="signed error", BG="background", FG="foreground".

$Kinect^{SL}$	$Kinect^{ToF}$
Ambient background light (Section 4.2)	
Below 1 μW: IP=0%, SD < 6 mm	Below 1 μmW: IP=0%, SD < 11 mm
Above 1 μmW: IP=100%	Below 10 μmW: IP=0%, SD < 30mm
	At 20 μmW: IP=0%, SD < 42 mm
Multi-device interference (Section 4.3)	
IP: < 16.3% (evenly distributed)	IP: < 22.7% (repetitively blocked)
Frame < 400: RMSE w/o interf. < 6.7 mm,	RMSE w/o interf. < 4.6 mm
RMSE w interf. < 7.7 mm	RMSE w interf. < 19.3 mm,
Rest of frames: RMSE w/o interf. < 5.8 mm,	
RMSE w interf. < 9.4 mm	
Temperature drift (Sec. 4.4)	
Before 10′: RMSE ∈ [4; 4.6] mm, SDA ∈ [0.6; 1.8] mm	RMSE ∈ [4.6; 5.3] mm & SDA ∈ [1.3; 1.5] mm
After 10′: RMSE rising from 4.0 to 7.1 mm, SDA ∈ [0.6; 1.0] mm	
Linearity error (Section 4.5.3); Pt#1 (center)...Pt#4 (corner); $Kinect^{SL}$ below 3 m	
SE1∈ [−34; −1.5], SE2∈ [−6.5; 62] mm,	SE1∈ [−8; 29] mm SE2∈ [−8; 17] mm,
SE3∈ [−4; 129] mm, SE4 ∈ [2.5; 76] mm	SE3∈ [−8; 37] mm, SE4∈ [−69; 62] mm
SD1-3 ∈ [0.4; 14] mm, SD4 ∈ [0.4; 28] mm	SD1-3 ∈ [0.8; 6.8] mm, SD4 ∈ [1.8; 90] mm
Systematic error: planarity (Section 4.5.4)	
SD(≤ 1.5 m) ∈ [1.2; 4.8] mm,	SD < 1.65 mm
SD(≤ 2.5 m) ∈ [2.7; 16.6] mm,	
SD(≤ 3.5 m) ∈ [7.5; 30.9] mm	
Intensity related error (Section 4.5.5)	
–not applicable–	distance error < 3 mm@ 1 m distance
Semitransparent Media & Scattering (Sec. 4.6)	
Light Penetration ≥ 80%:	
SE ∈ [1; 1.5] mm, IP < 5%	SE ∈ [17.89; 378.1] mm, IP < 2%
Multipath effect (Section 4.7)	
Incid. angle < 13°: IP > 90%,	Incid. angle < 10°: IP < 70%, Err. > 600 mm
Incid. angle > 20°: IP < 1%, Err. < 5 mm	Incid. angle ∈ [10°, 30°]: IP ≈ 100%
	Incid. angle > 30°: IP < 2%, Err. < 4 mm
Depth inhomogeneity and dynamic scenery (Section 4.8)	
Seg. "S0": FG from 52.5 to 99.48%,	Seg. "S0": FG from 53.2 to 44.8%,
SD < 5.9 mm, BG from 43.0 to 0.5%,	SD < 1.3 mm, BG from 35.9 to 19.5%,
SD < 6.0 mm, IP from 7 to 0%, SD < 0.9 mm	SD < 1.5 mm, IP from 10.9 to 35.6%,
Seg. "S1": FG from 52.33 to 100%,	SD < 2.1 mm
SD < 5.0 mm, BG from 19.4 to 0%,	Seg. "S1": FG from 57 to 41.11%,
SD < 3.5 mm, IP from 28.1 to 0%,	SD < 1.4 mm, BG from 36 to 18.3%,
SD < 3.1 mm	SD < 1.6 mm, IP from 6.8 to 40.4%,
	SD < 1.5 mm

halogen lamps. The radiosity on the wall depends on the number of active lamps and their distance to the wall. We measure the radiant emittance of the surface resulting from our light sources with a Newport 818-SL powermeter. The powermeter directly delivers power intensity in W/cm^2 and it is calibrated to 850 nm, which relates to the Kinect's laser diode illumination of 850 nm. An additional laser pointer allows for the directional adjustment of the powermeter's pipe to a point on the wall in order to accurately measure the radiance. To register the point at the wall with a pixel in the Kinect camera, we temporally attached a small cuboid as depth marker to the wall.

As both Kinect cameras have an NIR-filter suppressing visible light,[7] we equip the powermeter with an additional Kodak Wratten high-pass filter number 87C. This filter has a 50% cutoff at 850 nm. A pipe is mounted to the powermeter in order to measure the incident radiance for a specific spatial direction from single point on the wall.

We further add an Atik 4000 astronomy photography camera equipped with the same NIR filter alongside with a powermeter in order to verify radiance measurements provided by the powermeter setup. The astronomy camera measures the radiant emittance in a linear relation to the number of photons received per pixel, i.e. per observed direction. In our experiments we found a proper linear relation between both measurements (Fig. 8).

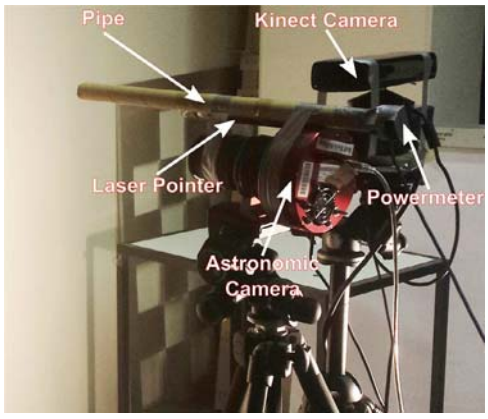

Fig. 8. Ambient background light experiment setup. The laser pointer is mainly hidden by the pipe.

We interrelate the radiance measurement of the powermeter to daylight condition. Therefore, we acquired a radiant flux density

[7] We did not explicitly measure the NIR filter, as this would require to destroy the Kinect camera.

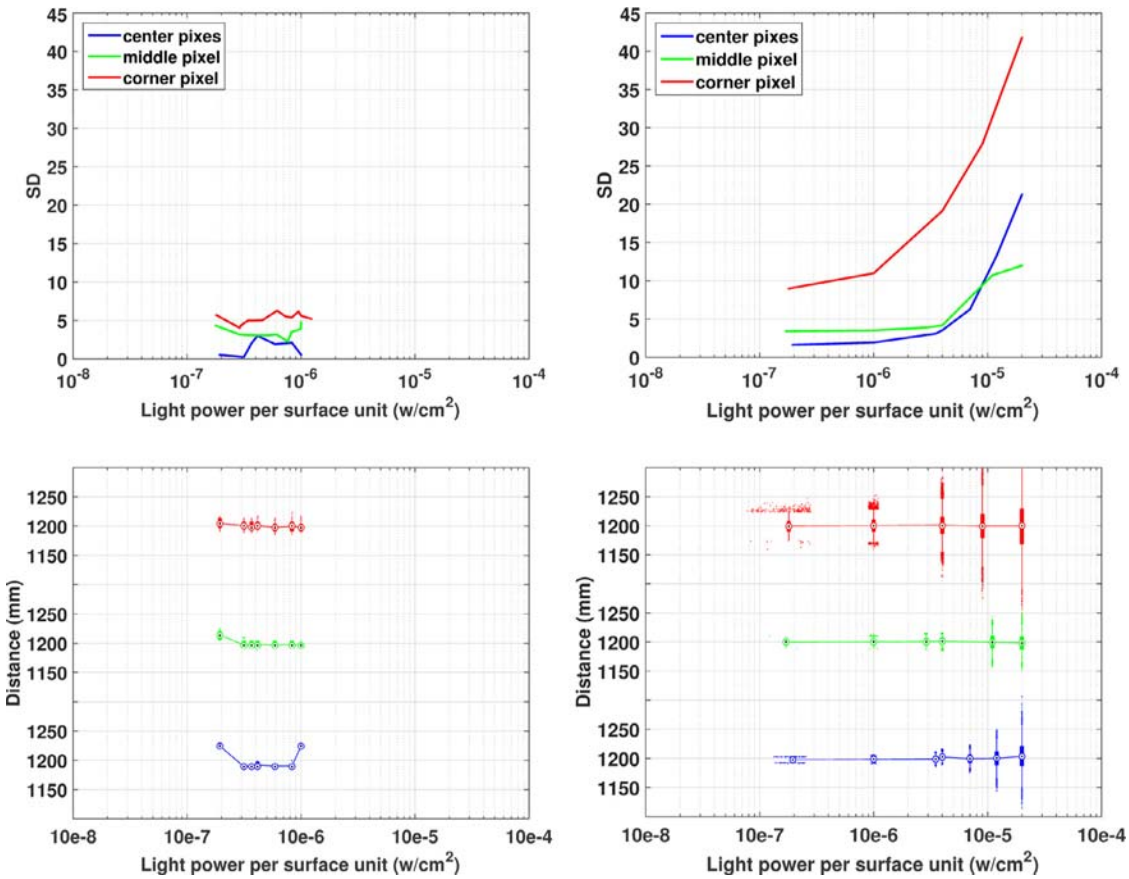

Fig. 9. Kinect comparison for ambient background light. The SD (top row) and distance statistics (bottom row) for the center, intermediate and corner pixel for Kinect^{SL} (left) and the Kinect^{ToF} (right).

reference measurement with the powermeter setup without pipe of direct sunlight on a sunny summer day in central Europe. This results in 11 mW/cm^2. Furthermore, we relate the radiant flux density measurement to the incident radiance measurement of an indirectly illuminated diffuse white paper using the powermeter with pipe at 1.2 m distance resulting in a factor of 1.1×10^3. As the later setup is comparable to the evaluation setup for the Kinect cameras, we can deduce a sun reference incident radiance value of about 10 μW/cm^2.

The final radiance measurements are done with the powermeter setup. The radiance measurements take place when the Kinect camera is turned off in order to prevent interference with the camera's illumination unit. We acquired 200 frames for various light conditions up to 20 μW/cm^2. Since we expect some variation of the effect for different pixel locations, we measured three points along the half diagonal, i.e. a point close to the upper left corner, the principal point of the range image and one intermediate point in between both points.

4.2.3. Evaluation and results

We apply a variance analysis for the $K = 200$ frames of range values $D_i(u, v)$, $i = 1, \ldots, K$ delivered by each camera for each of the pixels (u, v) (center, intermediate, corner) by computing the *Standard Deviation (SD)* over time

$$\mathrm{SD} = \sqrt{\frac{1}{K}\sum_{i=1}^{K}\left(D_i(u,v) - D^{\mathrm{mean}}(u,v)\right)^2},$$

$$D^{\mathrm{mean}}(u,v) = \frac{1}{K}\sum_{i=1}^{K} D_i(u,v), \qquad (1)$$

and plot this as function over the ambient light intensity; see Fig. 9, top. Additionally, Fig. 9, bottom, shows explicit distance values including box plots as function over ambient light.

It can be observed that the Kinect^{SL} is not able to handle background light beyond 1 μW, whereas the Kinect^{ToF} delivers range data throughout the full range of ambient light applied in the experiment.

The Kinect^{SL} delivers more robust depth values than the Kinect^{ToF} throughout the ambient background light range where valid data is delivered. All observed pixels are below 6 mm SD and the max. variation from the median is 25 mm for the corner pixel. The SD and the box plots show that for the Kinect^{SL} the depth variation is hardly effected by the ambient light, as long as valid range data is delivered. The plots for the different pixels show that the variation increases for pixels closer to the image vicinity.

The Kinect^{ToF}, at the other hand, shows the expected raise in the depth variation for increasing ambient light due to a reduced SNR. Whereas the center and the intermediate pixels show similar SD below 6 μW as the Kinect^{SL}, i.e. below 4 mm, the box plots reveal a larger number of outliers compared to the Kinect^{SL}. However, the Kinect^{ToF}'s corner pixel delivers worse SD and quantile statistics than the one for the Kinect^{SL}. In the range beyond 10 μW ambient light, the variation increases to some 22, 12 and 42 mm for the center, intermediate and corner pixels, respectively. The effect that the center pixel gets worse than the intermediate pixel may be explained by oversaturation effects solely due to the active illumination of the Kinect^{ToF}.

4.3. Multi-device interference

4.3.1. Goal

This experiment addresses the problem arising from the interference of the active illumination between several Kinect-cameras of the

Fig. 10. Multi-device interference experiment setup .

same type, when running them in parallel. Primarily, we want to evaluate the influence of the interference on the range measurement. Secondarily, we want to gain some insight into the temporal and spatial distribution of the artifacts.

Note that in contrast to other ToF-cameras (see Section 2.3) we are not able to modify the modulation frequencies for the Kinect^{ToF} in order to reduce or suppress multi-device interference artifacts.

4.3.2. Experimental setup

The general idea of the experiment is to acquire an approximately diffuse white planar wall, adding a second Kinect device of the same kind as interference over a longer period of time. As the Kinect^{SL} uses a static structured light pattern, a fully static setup may not capture the overall interference. As circumventing interference for the Kinect^{SL} may not always be possible with a "Shake'n'Sense"-like approach [7], we investigate the influence of the camera poses of the two devices on the interference. Thus, for the Kinect^{SL} setup, we mount the interfering device on a turntable and rotate it $\pm$ 10° about the vertical axis with 1 RPM in order to get a variation of the overlay of the SL-patterns of the two devices. The angular speed is low enough to prevent any motion artifacts. We also investigated different inter-device distances, but the resulting impact on the interference was comparable. The Kinect^{ToF} setup the interfering device is always static. The distance between the wall and the Kinect was set to 1.2 m and the distance between the devices is 0.5 m; see Fig. 10. We do not take the exact orientation of the measuring and the interference devices into account, the measuring and the interfering device, but both devices observe approximately the same region on the wall.

4.3.3. Evaluation and results

In order to account for a potential misalignment of the Kinect toward the wall, we use a RANSAC plane fit to the range data with inactive interference device. The per-pixel range values, deduced from this plane fit, are considered as reference distances $D^{ref}(u, v)$. We compute the deviation for each frame D_i in respect to the reference distance as Root-Mean-Square Error (RMSE), i.e.

$$\mathrm{RMSE} = \sqrt{\frac{1}{m \cdot n} \sum_{u=1}^{n} \sum_{v=1}^{m} \left(D_i(u, v) - D^{ref}(u, v) \right)^2} \tag{2}$$

where n, m represent the width and height of the averaged range image, respectively.

As can be seen in Fig. 11, the active frequency pattern of the Kinect^{ToF} has a stronger interference than the structured light pattern for the Kinect^{SL} for most poses of the interfering camera. On average, the Kinect^{SL} shows little interference effect (RMSE $<$ 5.6 mm), beside some very prominent poses (RMSE up to $<$ 9.4 mm). Fig. 12, mid-right, shows a sample range image with a high RMSE. The Kinect^{ToF} camera shows low interference for the majority of the frames (RMSE: $<$ 5 mm), but extreme interference errors for some 25% of the frames (RMSE up to 19.3 mm) that occur in a sequence which has a nearly

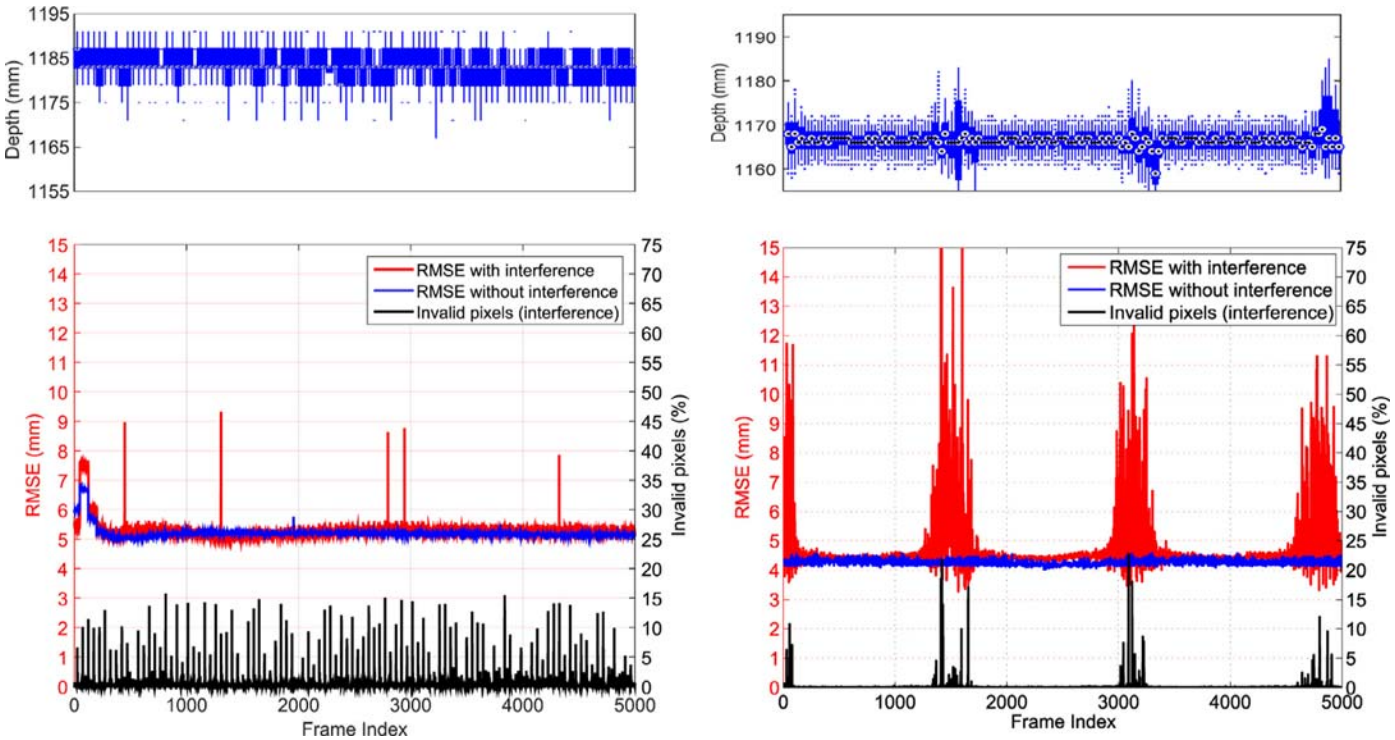

Fig. 11. Multi device interference: Error for the static scene for the Kinect^{SL} with moving interference device (left) and for the static Kinect^{ToF} (right) as box plot statistics for the interference situation (top row) and RMSE plot with and without interference including invalid pixel counts for the interference setup (bottom row).

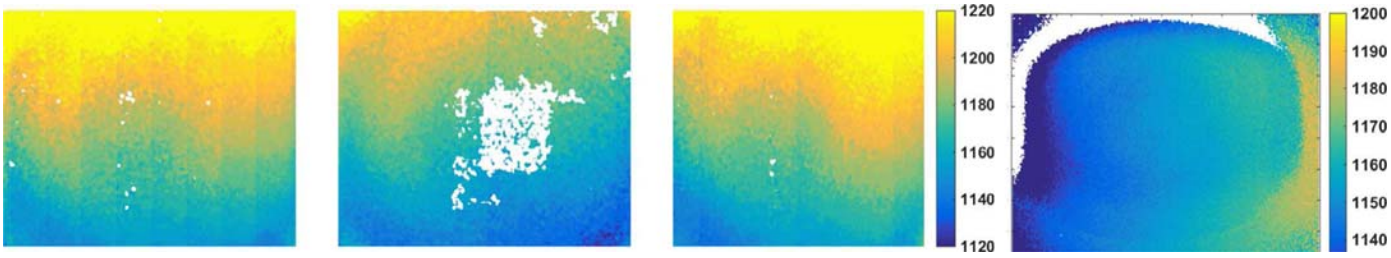

Fig. 12. Sample range images for the multi-device interference setup: The Kinect^{SL} range image for the initial pose of the interfering device, left, and for two pose with a high invalid pixels count, mid-left, and a high RMSE, mid-right. A Kinect^{ToF} range image with high invalid pixel count and high RMSE, right. Invalid pixels are represented as pure white.

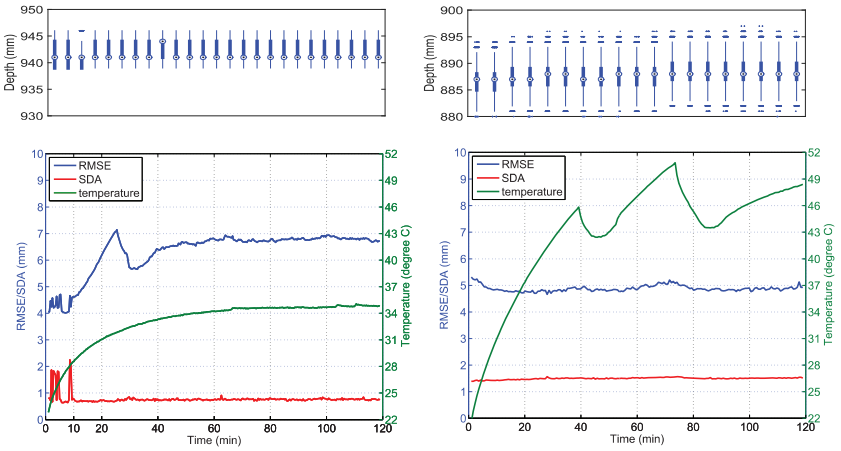

Fig. 13. Device warm up: Box plot error mean and temperature versus warm-up time (bottom row) for Kinect^{SL} (left) and Kinect^{ToF} (right).

constant repetition rate. This behavior is most likely due to the asynchronous operation of the two devices. A signal drift over time between the signals generated in both devices would lead to a repetitive interference pattern as the one observed. The range statistics represented in Fig. 11, top, shows that the median in Kinect^{SL} is not altered by interference, which is mainly due to the strong quantization applied in the disparity maps. In phases of maximum interference, the Kinect^{ToF} delivers increased drift of the median, up to 5 mm, and a stronger variation.

Regarding the invalid pixels, the Kinect^{SL} nearly always delivers invalid pixels. For the initial pose, we find some 1.5% invalid pixels; see Fig. 12, left. While changing the pose of the interfering device, we find up to 16.3% invalid pixels; see Fig. 12, mid-left. The Kinect^{ToF} does not deliver invalid pixels in the non-interfered periods, but in the interference periods up to 22.7% of invalid pixels have been observed; see Fig. 12, right.

We want to point out that we always observe strong variations within the first 400 frames, i.e. the first 13 s after starting the acquisition with the Kinect^{SL}. In this experiment we have an increased RMSE of up to 6.7 mm without interference and 7.7 mm with interference. Therefore, it is advisable to not use this initial sequence captured with the Kinect^{SL}.

4.4. Device warm-up

4.4.1. Goal

This test scenario is designed to evaluate the drift in range values during the warm-up in standard operation, i.e. the stability of the range measurements of both Kinects with respect to the operating time.

4.4.2. Experimental setup

We accommodate the device in a room with a constant temperature of 21 °C which is actively controlled by an air conditioning system with a variance below 0.1 °C. We start to operate the device measuring a planar wall at a distance of 1200 mm. We acquire 200 frames in a row and drop frames for 15 s (450 frames) afterward and repeat this until a total time of 120 min. During the acquisition a digital thermometer (precision ± 0.1 °C) records the temperature inside the Kinect devices. The temperature in the device interior is measured with a flexible sensor tip inserted through the ventilation holes. Thus, the devices remained intact in order to keep the original temperature dissipation system.

4.4.3. Evaluation and results

As the variation in the range data is smaller for the cold device than for the warm device, we make a RANSAC fit to the averaged first steady sequence of 200 frames, resulting in a reference depth image D^{ref}. However, for Kinect^{SL} the first steady sequence of 200 frames was captured after 10 min, as we observe a very strong variation in this initial range of measurements. Nevertheless, since the RANSAC is applied to the whole frame there might be some bias to the reference frame. We calculate the RMSE for the average of all 200 frames D^{mean} in a frame sequence with respect to the fitted plane as

$$\mathrm{RMSE} = \sqrt{\frac{1}{m \cdot n} \sum_{u=1}^{m} \sum_{v=1}^{n} \left(D^{\mathrm{mean}}(u,v) - D^{\mathrm{ref}}(u,v) \right)^2}. \tag{3}$$

Furthermore, we calculate the per-pixel *standard deviation average* (SDA) for each sequence of $K = 200$ frames D_i.

$$\mathrm{SDA} = \frac{1}{m \cdot n} \sum_{u=1}^{m} \sum_{v=1}^{n} \sqrt{\frac{1}{K} \sum_{i=1}^{K} \left(D_i(u,v) - D^{\mathrm{mean}}(u,v) \right)^2}. \tag{4}$$

The results for the device warm-up test are shown in Fig. 13. The fluctuation in the temperature of the Kinect^{ToF} is due to the cooling system, that gets activated and deactivated depending on the system temperature. For the Kinect^{SL} there is only a small temperature difference of 11 °C after 120 min. The results show that Kinect^{ToF} has

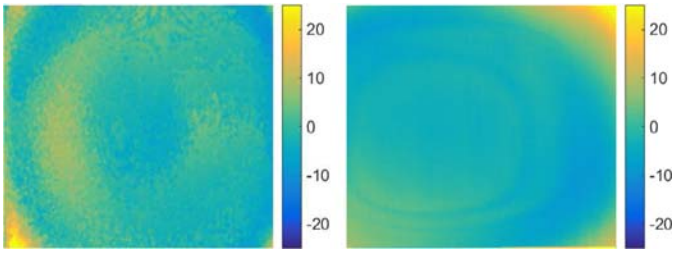

Fig. 14. Device warm up: Depth error at minute 60 for Kinect^{SL} (left) and Kinect^{ToF} (right).

in general less error than the Kinect^{SL} and SDA and RMSE are nearly constant over time. The Kinect^{SL} has strong error and variation fluctuations in the first 10 min of the warm-up phase. After the device has reached its operation temperature, the distance SDA stays within 1 mm, which is slightly better than for the Kinect^{ToF} which has a distance SDA of 1.5 mm. However, the distance error is higher for the Kinect^{SL} (RMSE $<$ 7.1 mm) than for the Kinect^{ToF} (RMSE $<$ 5.3 mm). The distance box plots in Fig. 13, top, show less variation for the Kinect^{SL} than for the Kinect^{ToF}. However, we again point out that the homogeneous appearance of the box-plots for the Kinect^{SL} partially result from the heavy quantization applied in this device.

Fig. 14 shows the depth error in respect to the fitted plane in absolute signed values. The depth images are taken at minute 60, when both devices are at a stable temperature. As it can be seen in the depth images the Kinect^{ToF} delivers smoother results with less out of plane errors compared to Kinect^{SL}, which is consistent with the RMSE values at minute 60.

4.5. Rail depth tracking

4.5.1. Goal

This test scenario primarily addresses the quality of the range data in respect with ground truth distances for a planar wall. The test involves the linearity including planarity tests as well as the intensity related error. The latter applies only for the Kinect^{ToF} camera. A secondary goal is to give some clue about the dependence of the error from the pixel location, therefore we evaluate the error at a few different image locations.

4.5.2. Experimental setup

The setup comprises a motorized linear rail mounted perpendicular to a white wall, which measures distances between 0.5 m and 5 m at a step-size of 2 cm. The camera is mounted on the carriage of the rail facing perpendicular to the wall. As the wall does not cover the full range image for farther distances, we evaluate planarity and linearity of the camera only within a region-of-interest including pixels lying on the white flat wall in the full distance range. The pixel region of interest for Kinect^{SL} is (1,1), (630,480) and (74,4), (502,416) for Kinect^{ToF}. Furthermore, we observe some pixels along a line-of-interest from the image center to the top-left corner, which are always covering the wall. We acquire 200 frames for each distance. For the evaluation of the intensity-related error, the acquisition is repeated with a 5 $\times$ 6 checkerboard attached to the wall. The checkerboard consists of 10 gray-level rectangles on white background, where the gray-level degrades from 100% to 0% black. The checkerboard has been printed using a standard laser printer which delivers sufficiently proportional reflectivity in the visual and the NIR range.

In order to re-project the range values into 3D-space, we first estimated the camera intrinsics using the well known photometric calibration technique from [57]. Similar approaches have been applied to the Kinect^{SL} [37] (where the laser beam is obstructed and an incandescent lamp is used to highlight the checkerboard in order to acquire a reliable NIR image of the calibration rig) and for ToF-cameras [33].

4.5.3. Linearity: evaluation and Rresults

The evaluation of the linearity requires a proper measurement of the ground truth distances for the range images acquired with the rail system. As a perfect orthogonal alignment of the camera toward the wall cannot be guaranteed, we propose to bypass this problem using photometric methods. Having a complete lens calibration of both camera systems (i.e. depth and color intrinsic and distortion parameters) and the extrinsic transformation between the High-Res color camera and the depth camera, the precise 3D camera position of the depth camera can be obtained using a simple black–white checkerboard reference fixed to the planar wall and which we acquire at 10 different rail positions. The corresponding 3D positions of the depth camera relative to the reference wall is done using the standard method [57]. A 3D line was fitted to these 3D positions using a RANSAC statistical approach which gives the robust orientation of the linear rail. Finally, knowing the precise displacement of each measurement of the linear rail (we use a 2 cm step size), the 3D position of the camera can robustly be estimated and thus a precise ground-truth of the wall be generated using the lens parameters of the depth camera. Having the ground truth distance $D_d^{gt}(u,v)$ for a given camera-to-wall distance d for each pixel, we calculate the *signed error (SE)* for the average of all depth measurements $D_d^{mean}(u,v)$ at the rail system, thus suppressing sensor noise

$$\mathrm{SE}_d = \frac{1}{k \cdot l} \sum_{u=1}^{k} \sum_{v=1}^{l} \left(D_d^{mean}(u,v) - D_d^{gt}(u,v) \right), \tag{5}$$

within the region-of-interest consisting of $k \times l$ pixel. For some pixels along the line-of-interest we also evaluate the individually signed linearity errors.

Furthermore we calculate the variance for each pixel in the region-of-interest using the 200 frames $D_{i,d}$ taken for each camera-to-wall distance d in order to retrieve the standard deviation average SDA according to Eq. (4).

Fig. 15, top, shows the signed linearity errors for both Kinects for the selected pixels with some box plots superimposed. As can be seen in Fig. 15 right, the Kinect^{ToF} delivers more precise range data than the Kinect^{SL}, if the corner pixel is not taken into account. In the proposed work range of the Kinect^{SL} below 3 m the error lies in the range of $[-34, -1.5]$ mm for the best (central) pixel and of [2.5, 76] mm for the worst (peripheral) pixel. The SD for the Kinect^{SL} below 3 m is very similar for all pixels and is below 30 mm. Above 3 m the distance error of the Kinect^{SL} strongly increases for peripheral pixels. Even though there seem to be some fluctuations in the distance error for the Kinect^{ToF}, this effect is much smaller and much less regular than the "wiggling"-error observed so far for ToF-cameras [33,35]. For pixels not in the extreme periphery the absolute per-pixel distance error and the SD lies in the range of $[129, -34]$ mm and [0.4, 14] mm, respectively. For the corner pixel the SD range increases to [0.4, 28] mm.

4.5.4. Planarity: evaluation and results

The region-of-interest in each range image acquired using the rail lies on a planar wall, so the resulting range measurements should ideally result in a plane. Similar to Khoshelham and Elberink [26], we apply a RANSAC plane fitting method to avoid outliers and calculate the standard deviation of the points from the fitted plane as planarity error.

Fig. 16 shows the planarity error as SD for both Kinect and the theoretical random error deduced by Khoshelham and Elberink [26] for the Kinect^{SL}. The Kinect^{SL} delivers much stronger out-of-plane errors than the Kinect^{ToF}, which stays below 1.65 mm for the whole range of 4 m. The curve for the Kinect^{SL} is roughly within the expected range.

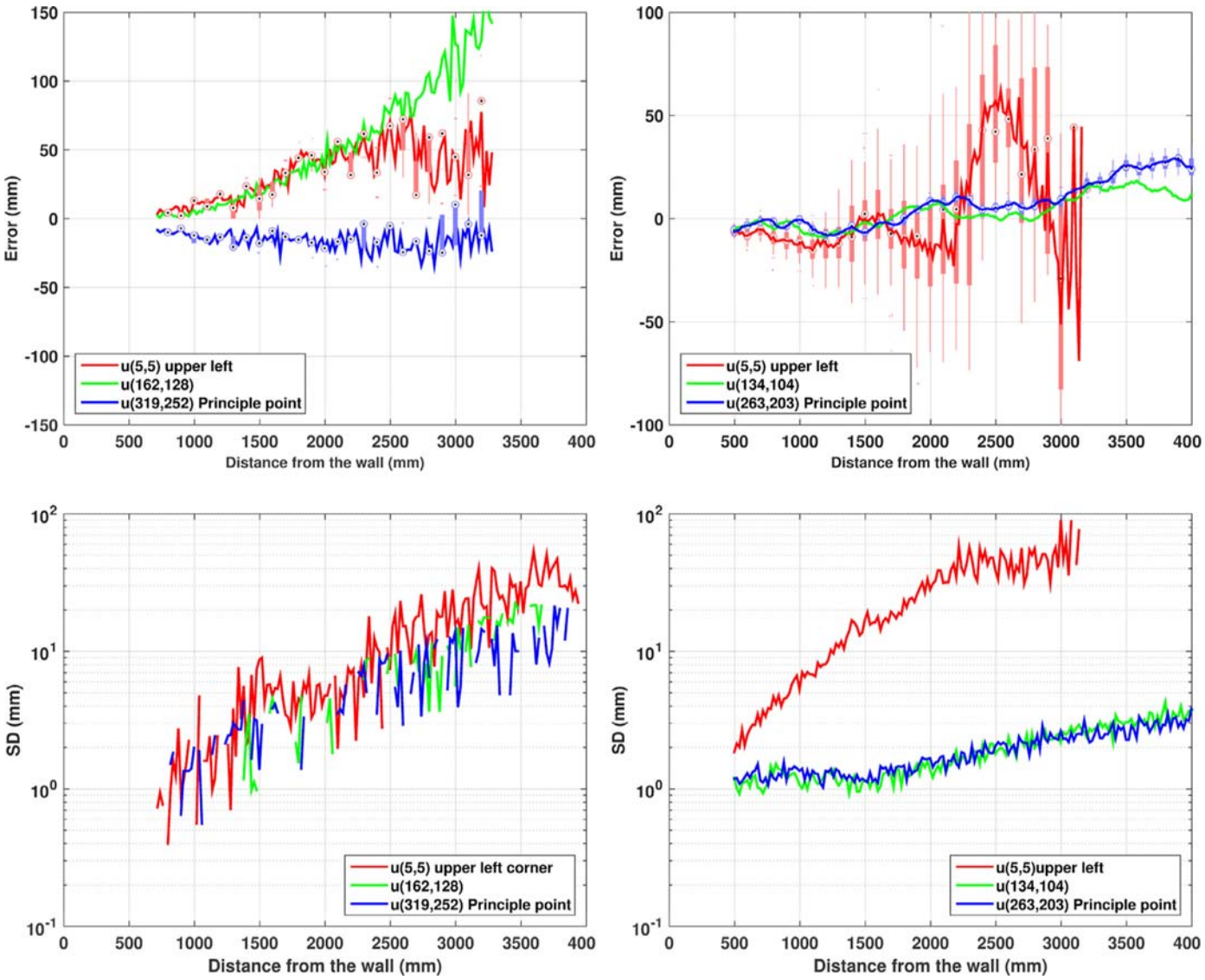

Fig. 15. Linearity error for four points along the line of interest: Distance error and SD of for Kinect^{SL} (left) and Kinect^{ToF} (right). The corner pixel of the Kinect^{ToF} delivers invalid depth after about 3100 mm, therefore no depth values are given for this range.

Compared to Khoshelham and Elberink [26] we observe an additional fluctuation which can be explained by the decreasing depth resolution of the Kinect^{SL} which leads to a significant depth quantization for increasing distances.

4.5.5. Intensity (Kinect^{ToF} only)

Similar to Lindner et al. [35] we evaluate the planar checkerboard with varying gray-levels at 1 m distance. For this scenario we select horizontal pixel lines across the gray-level rectangles and directly plot the distance values for several distances to the wall. Fig. 17 shows that the Kinect^{ToF} delivers very stable results and the range error for the darkest rectangle is max. 3 mm, compared to the white reference distance. Compared to earlier ToF-camera prototypes, for which range errors up to 50 mm have been observed [35], this is a significant improvement of quality.

4.6. Semitransparent liquid

4.6.1. Goal

This test scenario is designed in order to evaluate the effects of translucent, i.e. semitransparent and scattering material on the quality of the acquired object geometry.

4.6.2. Experimental setup

Similar to Hansard et al. [18] we use a sequence of semitransparent liquids, i.e. a plastic cylinder filled with diluted milk. The cylinder has an inner and outer diameter of 77 and 79 mm, respectively. By diluting the milk with the same amount of water in each step, we get sequence of 10 objects with an amount of 2^{-k}, $k = 0, \ldots, 9$, i.e. 100%, … ,0.19% milk. We acquire 200 frames for each setup. The cylinders are acquired from frontal view at a distance of 1.2 m.

Fig. 18, bottom, the visual appearance of the milk probes is shown. The diluted milk is filled in cuvettes of 1 cm square cross section and placed in front of a checkerboard in order to demonstrate the degree transparency in the visual range.

In order to provide a quantitative transparency degree, we measured the light penetration through the cuvettes at 850 nm. The measured intensity through a cuvette filled with water was the reference (I_0) and each sample was divided by the reference. The Kodak Wratten 850 nm filter explained in Section 4.2 was applied to filter visible light.

$$\text{Penetration}_{@850\text{nm}} = \frac{I}{I_0} \times 100,$$

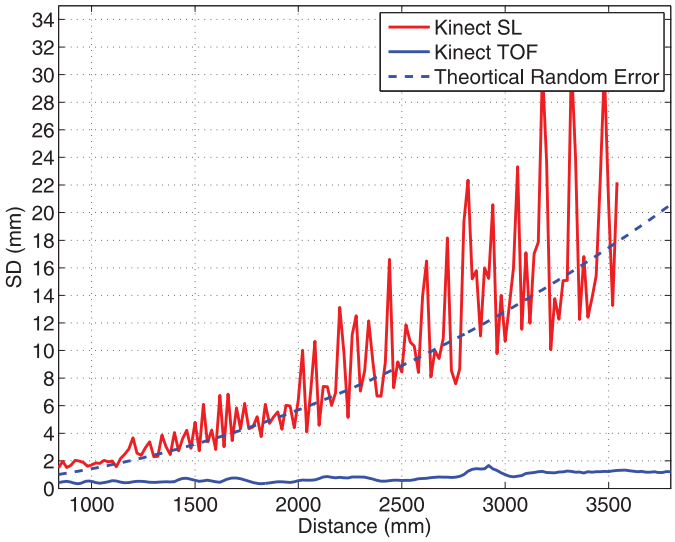

Fig. 16. Planarity error. The standard deviation of the pixels within the region-of-interest for the Kinect^{SL} (red) and the Kinect^{ToF} (blue). Additionally, the theoretical random error deduced by Khoshelham and Elberink [26] for the Kinect^{SL} is shown. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

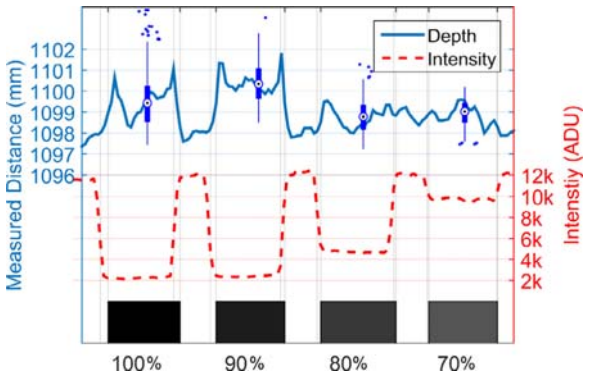

Fig. 17. Intensity related error for Kinect^{ToF}. Measured depth and intensity versus the actual intensity of the checker board at about 1 m distance. The intensity is given in *arbitrary digital units (adu)* as delivered by the Kinect^{ToF}.

4.6.3. Evaluation and results

Same as Hansard et al. [18] we directly measure the signed error in depth for a manually segmented region in the range image with respect to the mean image d^{mean} as a function of transparency by comparing against a reference measurement with a non-transparent cylinder of the same size; see Eq. (5). Furthermore, we plot the number of invalid pixels.

As can be seen in Fig. 18, top-left, Kinect^{SL} performs very well for liquid samples with more than 3.12% milk, with almost no invalid pixels and a signed error in the range of [1, 1.5] mm, which is around the thickness of the plastic cylinder. However, for the samples with concentration of milk below 3.12%, the number of invalid pixels increases dramatically to above 90% and the depth error of the remaining valid pixels is increasing as well. For the same experiments the Kinect^{ToF} shows a positive distance error between 12 and 378 mm, but does not mark any measurements as invalid, i.e. the number of invalid pixels is negligible; see Fig. 18, top-right. However, for the samples thinner than 3.1%, the number of invalid pixels increases dramatically to above 90% for the Kinect^{SL} while Kinect^{ToF} still delivers valid pixels with rising error up to 400 mm for 0.2% milk. In conclusion, Kinect^{SL} performs good for thicker semitransparent liquids and indicates failure for the thinner cases. On the other hand, using the Kinect^{ToF} is much harder, as the device does not indicate the pixel's invalidity even for a large amount of distance error.

4.7. Reflective board

4.7.1. Goal

This test evaluates the impact of strongly reflecting objects which potentially result in erroneous depth measurements mainly due to multi-path effects. Beside the reflectivity as such, the multi-path effect strongly depends on the orientation of the reflective object towards other bright objects in the scene and the camera. Therefore, we are mainly interested in the relation between the angular orientation and measured depth error.

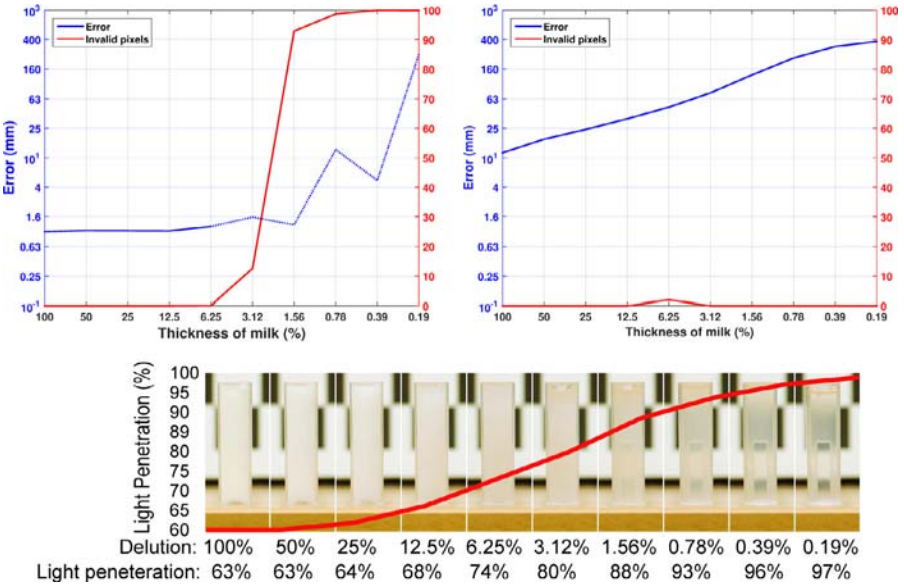

Fig. 18. Semitransparent liquid. Depth error and amount invalid pixels versus the transparency of the liquid for the Kinect^{SL} (left) and the Kinect^{ToF} (right). Samples in standard cuvette are shown in bottom row.

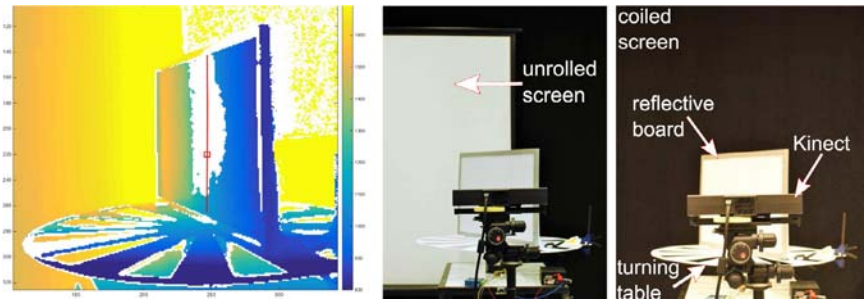

Fig. 19. Reflective board. Sample range image with imposed pivot-line (left) and a photo of the setup with unrolled screen (middle) and coiled curtain (right).

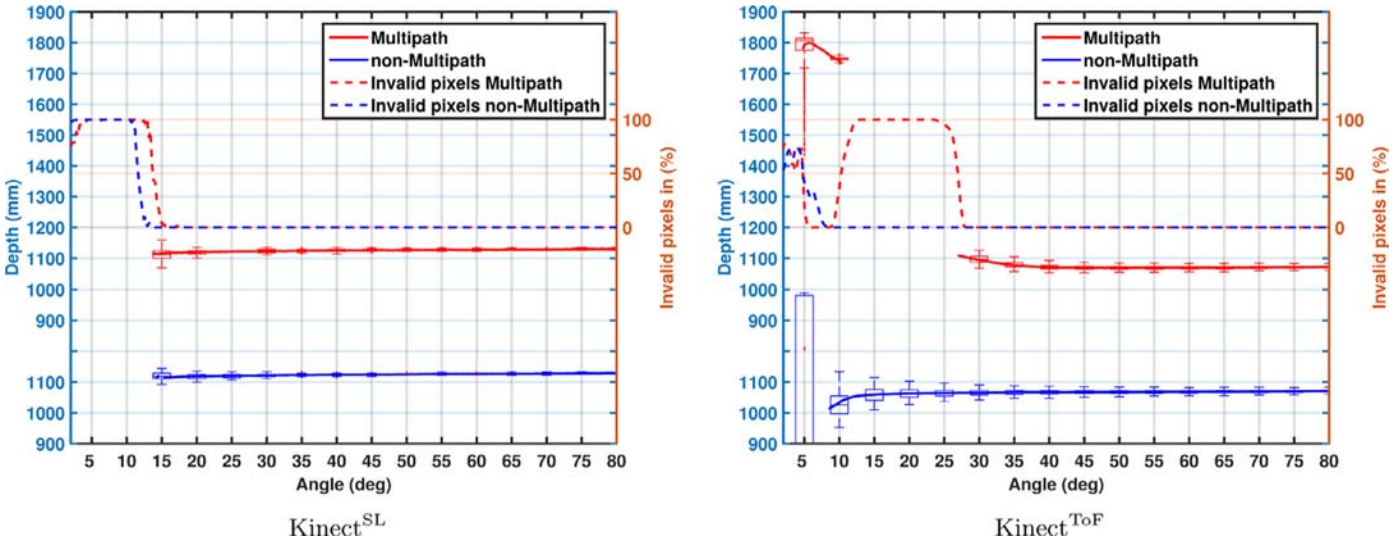

Fig. 20. Reflective board. The SD and the amount of invalid pixels versus the angle of incidence for the non-multi-path and the multi-path situation for the Kinect^{SL} (left) and the Kinect^{ToF} (right).

4.7.2. Experimental setup

We use a common whiteboard of 60 × 40 cm size as reflective object and place it vertically on a turning table in front of a white projector screen at a distance of 170 cm from the camera. The projector screen can be rolled up and behind that there is a non-reflecting black curtain in order to make a non-multi-path reference measurements; see Fig. 19, middle and right. The rotating vertical board is placed in front of the Kinect camera so that the board rotates around the pivot line which intersects the center of the rotating table. The points lying on the pivot line remain at the same distance to the camera. The rotation starts from 0° to 90° with resolution of 0°15′. The specific multi-path effect depends on the board angle. For each step we acquire 20 frames.

4.7.3. Evaluation and results

For each acquired pair of range images, i.e. for a given fixed angle, with a coiled and unrolled screen, we select a vertical 4 × 100 pixel region of interest around the rotation pivot on the whiteboard. For each pixel in the vertical region of interest we assume a constant distance to the camera and a constant multi-path situation. Within this region we compute the RMSE with respect to the reference measurement at 90° and the SD of the measurement itself as a function of the incident angle. Furthermore, we calculate the relative number of invalid pixels.

In Fig. 20 the RMSE and the SD for all acquisition angles for the Kinect^{SL} (left) and the Kinect^{ToF} (right) are plotted. Additionally, we plot the amount of invalid pixel. As expected, the Kinect^{SL} has much less problem with this indirect lighting setup, since the structured light principle does not get confused by diffuse scattered light. However, the Kinect^{SL} has also limitations for low angles and delivers a higher invalid depth for low incident angles. Even though the measurement principle should not get affected by this. One simple explanation would be, that too little light is getting reflected to the camera, however, this would also be true for the reference measurement with coiled screen.

For incident angles below 10° up to 100% of the pixels are marked invalid. For angles above 15°, the Kinect^{SL} yields nearly no invalid pixels and the depth error is close to zero. The Kinect^{ToF}, on the other hand, has a lot more problems with the superposition of the indirect illumination, i.e. the multi-path situation. Apparently, the Kinect^{ToF} is able to detect some of the corrupted pixel, but at angles below 10° which get affected by a low incident angle, are not classified as invalid, resulting in extremely range errors up to 800 mm. For incident angles between 10° and 30° the Kinect^{ToF} delivers up to 100% invalid pixel. Similar as for the Kinect^{SL}, the Kinect^{ToF} range values are again more reliable for angles above 35°, i.e. no invalid pixels are delivered with a depth error below 50 mm.

4.8. Turning Siemens star

4.8.1. Goal

The performed test targeted at measuring the amount of flying pixels, i.e. pixels that cover an inhomogeneous region in terms of depth and thus do not deliver proper depth values, for static and dynamic scenes. Both Kinect cameras mark unreliable pixels as “invalid”, which also applies for the flying pixels.

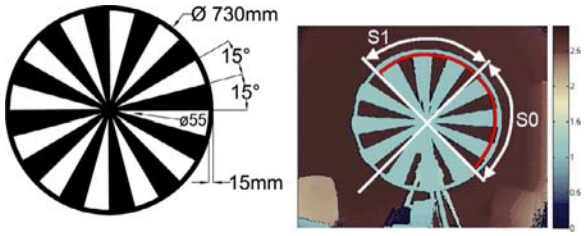

Fig. 21. The "Siemens star". Mechanical details (left) and the segments as well as the pixels used for the evaluation are marked in red (right).

4.8.2. Experimental setup

Similar to Lottner et al. [36] we manufactured a 3D Siemens star. However, we mount it to a stepping motor that actuates the star in a controlled fashion in front of a planar background wall at 1.8 m distance. The geometrical dimensions of the Siemens star are shown in Fig. 21, left. We apply different angular velocity while capturing range data with any of the Kinect cameras. As the both Kinects have different intrinsic parameters and different range image resolution, there are different options for the geometric setup of the measurement. We opted for a setup where each sensor has the same "pixel coverage" on the star, thus the cameras have different distances to the star during the acquisition, but as both cameras have approximately the same temporal resolution, i.e. frame rate, pixel coverage for the lateral motion is comparable for both Kinect.

We acquire range data for the static and the dynamic wheel. We have chosen nine velocity steps between $0, 11, 22, \ldots, 100$ RPM. In our setup 100 RPM relates to 35 pixel swept in the most outer circle (red arc in Fig. 21, right) by the wheel within one range image, i.e. in 1/30 s.

4.8.3. Evaluation and results

In the evaluation we account for the fact, that the Kinect's illumination units are mounted horizontally for both cameras, leading to different shadowing effects at vertical and horizontal edges. Thus we expected varying results between regions with predominantly vertical and horizontal edges and performed the analysis separately for the two wheel quarters, one at the right ("S0") and one at the top ("S1"). For the evaluation we use pixels at a circular arc at the outer part of the wheel illustrated by the red arc; see in Fig. 21, right.

In an ideal case, along the arcs there should be 50% foreground and 50% background pixels. Therefore we simply calculate the minimal, mean and maximal relative numbers of foreground, background and invalid pixels for the different speed values.

Fig. 22 shows the results of the foreground-background analysis for both cameras and both segments. One first insight is, that the classification results are very stable, namely the Kinect^{ToF} shows very little variation in its results. Comparing the classification results for

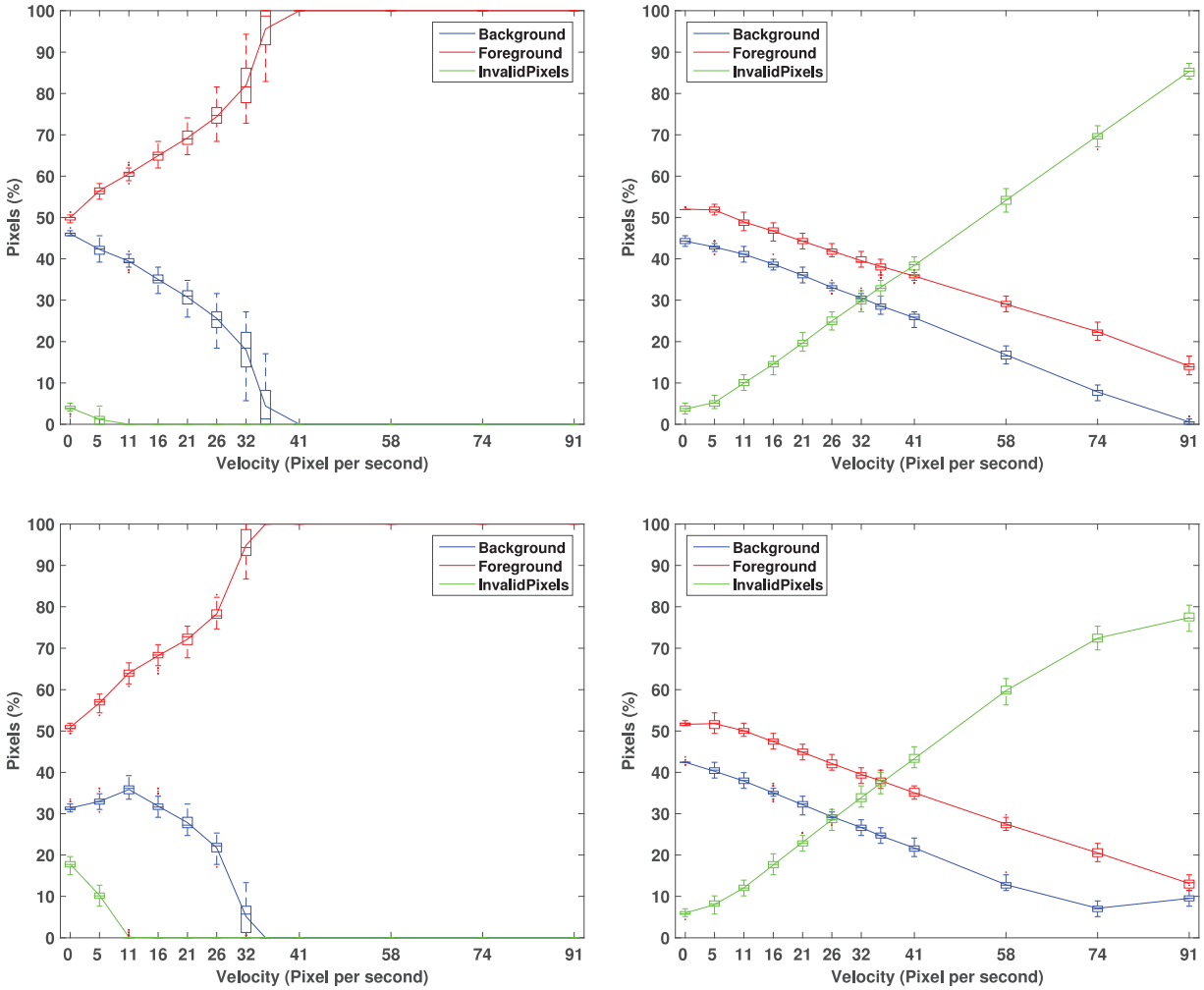

Fig. 22. Analysis for the turning Siemens star. The minimal, mean and maximal relative numbers of foreground, background and invalid pixels plotted over the angular velocity for the Kinect^{SL} (left) and the Kinect^{ToF} (right) for the segments "S0" (top) and "S1" (bottom).

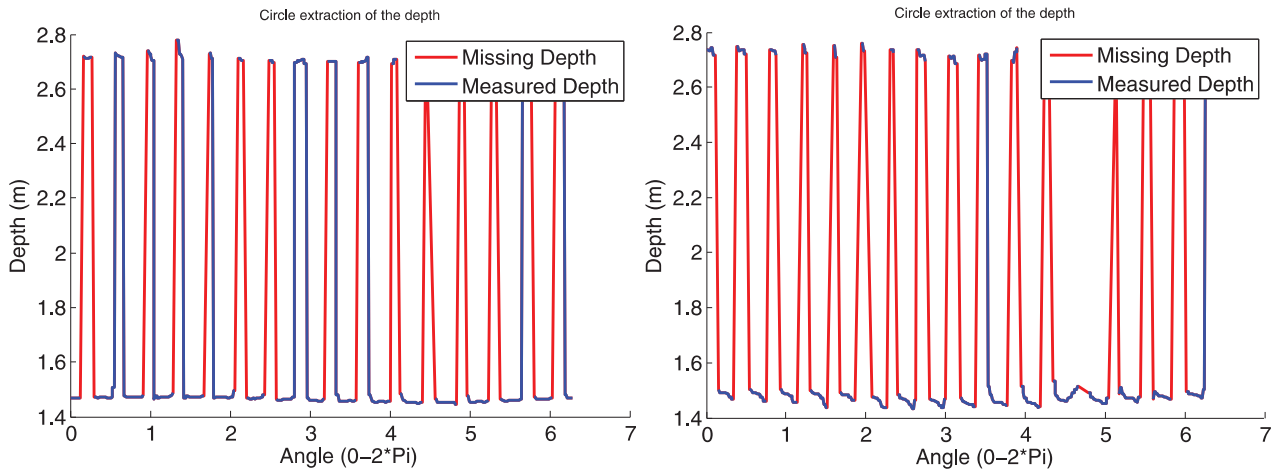

Fig. 23. Range Profile for the turning Siemens star for the angular speed of 0RPM (left) and 60RPM (right) measured by Kinect^{ToF}. Red indicates invalid pixels. Note that the range between 4.3 and 5 radians is the lower part of the wheel where the tripod distorts the background region.

the static scene, i.e. the flying pixels, the foreground classification is nearly perfect, i.e. 52.5% for the Kinect^{SL} and 53.2% for the Kinect^{ToF} for segment "S0". The amount of invalid pixel for this segment is 7.0% for the Kinect^{SL} and 10.9% for the Kinect^{ToF}.

For increasing speed, it is apparent that the Kinect^{SL} delivers less invalid pixels and more (false) foreground pixels, resulting in 100% foreground pixels for 100 RPM. The behavior of the Kinect^{ToF} is much more reliable. As one would expect, the number of invalid pixels increases for higher speed and the number of foreground and background pixel decreases in a comparable way. However, there are always more foreground than background pixel. This effect can be explained by the shadowing which only applies to the background, i.e. the holes in the Siemens star.

As expected, there are differences between the two arc segments for both Kinect. In general, the results for the top segment "S1" are worse for both devices, as shadowing effects are stronger for vertical edges. For the Kinect^{SL} mainly the number of invalid pixels is higher for lower speed, which is counter-intuitive. For the Kinect^{ToF} the differences between the two segments are less prominent.

Beside the classification for the pixels along the arc, Fig. 23 shows the range profile for 0 RPM and 60 RPM for the full outer pixel circle. This profile plot shows an additional range distortion effect at the edges of the foreground parts. Here, additional "overshooting" effects occur, which are due to motion artifacts apparent to ToF cameras; see Section 2.3.

5. Conclusion

This paper presents an in-depth comparison between the two versions of the Kinect range sensor, i.e. the Kinect^{SL}, which is based on the Structured Light principle, and the new Time-of-Flight variant Kinect^{ToF}. We present a framework for evaluating Structured Light and Time-of-Flight cameras, such as the two Kinect variants, for which we give detailed insight here. Our evaluation framework consists of seven experimental setups that cover the full range of known artifacts for these kinds of range cameras.

5.1. Device selection hints

Since device selection is highly application dependent, Table 5 sets up some rules of thumb to help users to make a more profound decision on which device to select depending on their application circumstances. The table compares device performance in different conditions with respect to the main error sources discussed above. For each source, we define two modes of operation, i.e. two ranges of application parameters. For each mode, we state a failure weight which is the performance ratio for both Kinect cameras. The individual failure values are deduced by combining error values and the number of valid pixels for each mode. Based on the individual failure values we compute the ratio between the Kinect^{SL} and the Kinect^{ToF} failure, whereby a ratio close to 1 means that both devices perform

Table 5
Device failure ratios for two application modes for the major error sources discussed in this paper. A ratio of 1 indicates equal behavior of both devices, values close to 0 and infinity indicate high relatively failure for Kinect^{SL} and Kinect^{ToF} respectively. The color intensity indicates deviations from 1, i.e. cases where both devices behave differently.

	Failure ratio (KinectSL / KinectTOF)	
Ambient Background Light	Below 1μW/cm2	Above 1μW/cm2
	0,55	**INF. ∞**
Depth Inhomogeneity and Dynamic Scenery	Static	Dynamic>40 pixel/s
	4,00	**INF. ∞**
Semitransparent Media & Scattering	Light penetration<80%	Light penetration>80%
	0,16	**INF. ∞**
Multipath Effect	15<Angle<25	25<Angle
	0,00	**1,00**
Linearity Error	< 2.5 m	> 2.5 m
	7,00	**5,50**
Systematic Error: Planarity	< 2.5 m	> 2.5 m
	5,45	**12,12**
Temperature Drift	Before 10'	After 10'
	0,87	**1,52**

quite similarly, whereas values close to 0 or close to infinity indicate, that Kinect^{SL} and Kinect^{ToF} have relatively high failure rates, respectively. For a specific application scenario, the user selects relevant error sources and by multiplying the failure ratios, the overall failure ratio is computed. If this final ratio is smaller than 1, Kinect^{SL} would be the best choice, otherwise Kinect^{ToF} is preferable. Of course, this is only a very coarse but quick guideline resulting in a first suggestion. The user should in any case have a further look at the details for the error sources that are most relevant to the specific application.

Note that we dropped the error sources "Intensity related error" and "Multi-device interference" from Table 4, because the "Intensity related error" applies only to Kinect^{ToF} and has compared to prior ToF devices only very little impact. Furthermore, if "Multi-device interference" is essential to the application, further actions need to be applied, such as using "Shake'n'Sense" in case of Kinect^{SL} [7] or different modulation frequencies in case of the Kinect^{ToF}.

Example 1. User A requires a depth sensing device for indoor scene reconstruction where the scene has static semi reflective surfaces at high angles:

Failure ratio $= 0.55 \times 4 \times 0.16 \times 0 \times 7 \times 5.45 \times 0.87 = 0$

Therefore Kinect^{ToF} would absolutely fail in this application.

Example 2. User B requires face gesture recognition at 1.2 m distance in indoor office conditions:

Failure ratio $= 0.55 \times 4 \times 0.16 \times 1 \times 7 \times 5.45 \times 0.87 = 11.68$

As the failure ratio is more than 1, user B should choose Kinect^{ToF} for his application.

5.2. Open science

We have prepared a website to make the following material publicly available:

1. A documented version of the evaluation scripts for all experiments written in Matlab.
2. Further technical details for setting up the required test scenarios, e.g. a CAD file for the Siemens star, intensity and calibration checker board.

The website is available for use by other researchers at: http://www.cg.informatik.uni-siegen.de/data/KinectRangeSensing/.

Acknowledgments

This research was partially funded by our collaboration partner Delphi Deutschland GmbH. The authors would like to thank Microsoft Inc. for making the prototype of the Kinect^{ToF}-cameras available via the Kinect For Windows Developer Preview Program (K4W DPP) and Dr. Rainer Bornemann from the Center for Sensor Systems of Northrhine-Westphalia (ZESS), Siegen, for the reference measurements of the illumination signal for the Kinect^{ToF} camera and for the support in measuring the ambient illumination.

References

[1] S. Bauer, A. Seitel, H. Hofmann, T. Blum, J. Wasza, M. Balda, H.-P. Meinzer, N. Navab, J. Hornegger, L. Maier-Hein, Real-time range imaging in health care: A survey, in: M. Grzegorzek, C. Theobalt, R. Koch, A. Kolb (Eds.), Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications, Lecture Notes in Computer Science, 8200, Springer, 2013, pp. 228–254.

[2] C. Beder, B. Bartczak, R. Koch, A comparison of PMD-cameras and stereo-vision for the task of surface reconstruction using patchlets, in: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2007, IEEE, 2007, pp. 1–8.

[3] K. Berger, S. Meister, R. Nair, D. Kondermann, A state of the art report on kinect sensor setups in computer vision, in: M. Grzegorzek, C. Theobalt, R. Koch, A. Kolb (Eds.), Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications, Lecture Notes in Computer Science, 8200, Springer, 2013, pp. 257–272.

[4] K. Berger, K. Ruhl, C. Brümmer, Y. Schröder, A. Scholz, M. Magnor, Markerless motion capture using multiple color-depth sensors, in: Proceedings of the Vision, Modeling and Visualization (VMV), 2011, pp. 317–324.

[5] J. Blake, F. Echtler, C. Kerl, OpenKinect: open source drivers for the kinect for windows v2 device, 2015 (https://github.com/OpenKinect/libfreenect2). Last visited: 26.01.15.

[6] G.R. Bradski, The OpenCV library, Dr. Dobb's J. Software Tools 25 (11) (2000) 120–126.

[7] A. Butler, S. Izadi, O. Hilliges, D. Molyneaux, S. Hodges, D. Kim, Shake'n'sense: reducing structured light interference when multiple depth cameras overlap, in: Proceedings of the Human Factors in Computing Systems (ACM CHI), ACM, New York, NY, USA, 2012.

[8] A. Dorrington, J. Godbaz, M. Cree, A. Payne, L. Streeter, Separating true range measurements from multi-path and scattering interference in commercial range cameras, in: Proceedings of the IS&T/SPIE Electronic Imaging, 2011, pp. 786404-1–786404-10.

[9] D. Droeschel, D. Holz, S. Behnke, Multi-frequency phase unwrapping for time-of-flight cameras, in: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2010, pp. 1463–1469.

[10] R. El-laithy, J. Huang, M. Yeh, Study on the use of Microsoft kinect for robotics applications, in: Proceedings of the IEEE Symposium on Position Location and Navigation (PLANS), 2012, pp. 1280–1288.

[11] G. Evangelidis, M. Hansard, R. Horaud, Fusion of range and stereo data for high-resolution scene-modeling (2015).

[12] D. Falie, V. Buzuloiu, Distance errors correction for the time of flight (ToF) cameras, in: Proceedings of European Conference on Circuits and Systems for Communications, 2008, pp. 193–196.

[13] P. Fechteler, P. Eisert, J. Rurainsky, Fast and high resolution 3D face scanning, in: International Conference on Image Processing (ICIP), 3, 2007, pp. 81–84.

[14] D. Fiedler, H. Müller, Impact of thermal and environmental conditions on the kinect sensor, in: Advances in Depth Image Analysis and Applications, Springer, 2013, pp. 21–31.

[15] L. Gallo, A.P. Placitelli, M. Ciampi, Controller-free exploration of medical image data: experiencing the kinect, in: Proceedings of the IEEE International Symposium on Computer-Based Medical Systems (CBMS), IEEE, 2011, pp. 1–6.

[16] O. Hall-Holt, S. Rusinkiewicz, Stripe boundary codes for real-time structured-light range scanning of moving objects, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2, 2001, pp. 359–366.

[17] J. Han, L. Shao, D. Xu, J. Shotton, Enhanced computer vision with Microsoft Kinect sensor: a review, IEEE Trans. Cybern. 43 (5) (2013) 1318–1334.

[18] M. Hansard, S. Lee, O. Choi, R. Horaud, Time-of-Flight Cameras: Principles, Methods, and Applications, Springer, 2013.

[19] D. Herrera C., J. Kannala, J. Heikkilä, Joint depth and color camera calibration with distortion correction, IEEE Trans. Pattern Anal. Mach. Intell. 34 (10) (2012) 2058–2064.

[20] T. Högg, D. Lefloch, A. Kolb, Real-time motion artifact compensation for PMD-ToF images, in: Proceedings of the Workshop on Imaging New Modalities, German Conference of Pattern Recognition (GCPR), in: LNCS, 8200, Springer, 2013, pp. 273–288.

[21] I. Ihrke, K.N. Kutulakos, H. Lensch, M. Magnor, W. Heidrich, Transparent and specular object reconstruction, in: Computer Graphics Forum, 29, 2010, pp. 2400–2426.

[22] A. Kadambi, R. Whyte, A. Bhandari, L. Streeter, C. Barsi, A. Dorrington, R. Raskar, Coded time of flight cameras: sparse deconvolution to address multipath interference and recover time profiles, ACM Trans. Graphics (TOG) 32 (6) (2013) 167.

[23] T. Kahlmann, F. Remondino, S. Guillaume, Range imaging technology: new developments and applications for people identification and tracking, in: Proceedings of Videometrics IX - SPIE-IS&T Electronic Imaging, 6491, 2007, doi:10.1117/12.702512.

[24] T. Kahlmann, F. Remondino, H. Ingensand, Calibration for increased accuracy of the range imaging camera SwissRanger^{TM}, Image Eng. Vision Metrol. (IEVM) 36 (3) (2006) 136–141.

[25] M. Keller, D. Lefloch, M. Lambers, S. Izadi, T. Weyrich, A. Kolb, Real-time 3D reconstruction in dynamic scenes using point-based fusion, in: Proceedings of Joint 3DIM/3DPVT Conference (3DV), 2013, p. 8.

[26] K. Khoshelham, S.O. Elberink, Accuracy and resolution of kinect depth data for indoor mapping applications, Sensors 12 (2) (2012) 1437–1454, doi:10.3390/s120201437.

[27] Y.M. Kim, C. Theobalt, J. Diebel, J. Kosecka, B. Miscusik, S. Thrun, Multi-view image and ToF sensor fusion for dense 3D reconstruction, in: Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops), IEEE, 2009, pp. 1542–1549.

[28] A. Kolb, E. Barth, R. Koch, R. Larsen, Time-of-flight cameras in computer graphics, Comput. Graphics Forum 29 (1) (2010) 141–159.

[29] K. Konolige, P. Mihelich, OpenKinect: Ros' technical description of kinect calibration, 2012 (http://wiki.ros.org/kinect_calibration/technical). Last edited: 27.12.12.

[30] K. Kuhnert, M. Stommel, Fusion of stereo-camera and PMD-camera data for real-time suited precise 3D environment reconstruction, in: Intelligent Robots and Systems (IROS), 2006, pp. 4780–4785.

[31] B. Langmann, K. Hartmann, O. Loffeld, Depth camera technology comparison and performance evaluation., in: International Conference on Pattern Recognition Applications and Methods (ICPRAM), 2012, pp. 438–444.

[32] D. Lefloch, R. Nair, F. Lenzen, H. Schäfer, L. Streeter, M. Cree, R. Koch, A. Kolb, Technical foundation and calibration methods for time-of-flight cameras, in: M. Grzegorzek, C. Theobalt, R. Koch, A. Kolb (Eds.), Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications, Lecture Notes in Computer Science, 8200, Springer, 2013, pp. 3–24.
[33] M. Lindner, A. Kolb, Lateral and depth calibration of PMD-distance sensors, in: Proceedings of the International Symposium on Visual Computing, in: LNCS, Springer, 2006, pp. 524–533.
[34] M. Lindner, A. Kolb, Compensation of motion artifacts for Time-of-Flight cameras, in: Proceedings of Dynamic 3D Imaging, in: LNCS, 5742, Springer, 2009, pp. 16–27.
[35] M. Lindner, I. Schiller, A. Kolb, R. Koch, Time-of-flight sensor calibration for accurate range sensing, Comput. Vision Image Understanding 114 (12) (2010) 1318–1328.
[36] O. Lottner, B. Langmann, W. Weihs, K. Hartmann, Scanning 2D/3D monocular camera, in: Proceedings of 3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, 2011, pp. 1–4.
[37] R. Macknojia, A. Chávez-Aragón, P. Payeur, R. Laganière, Calibration of a network of kinect sensors for robotic inspection over a large workspace, in: IEEE Workshop on Robot Vision (WORV), IEEE, 2013, pp. 184–190.
[38] S. Meister, S. Izadi, P. Kohli, M. Hämmerle, C. Rother, D. Kondermann, When can we use KinectFusion for ground truth acquisition? in: Proceedings of the Workshop on Color-Depth Camera Fusion in Robotics, 2012.
[39] R. Nair, S. Meister, M. Lambers, M. Balda, H. Hofmann, A. Kolb, D. Kondermann, B. Jähne, Ground truth for evaluating time of flight imaging, in: M. Grzegorzek, C. Theobalt, R. Koch, A. Kolb (Eds.), Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications, Lecture Notes in Computer Science, 8200, Springer, 2013, pp. 52–74.
[40] R. Nair, K. Ruhl, F. Lenzen, S. Meister, H. Schäfer, C. Garbe, M. Eisemann, M. Magnor, D. Kondermann, A survey on time-of-flight stereo fusion, in: M. Grzegorzek, C. Theobalt, R. Koch, A. Kolb (Eds.), Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications, Lecture Notes in Computer Science, 8200, Springer, 2013, pp. 105–127.
[41] R.A. Newcombe, A.J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, A. Fitzgibbon, Kinectfusion: real-time dense surface mapping and tracking, in: Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR), 2011, pp. 127–136.
[42] C.V. Nguyen, S. Izadi, D. Lovell, Modeling kinect sensor noise for improved 3D reconstruction and tracking, in: 2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), IEEE, 2012, pp. 524–530.
[43] M. Nießner, M. Zollhöfer, S. Izadi, M. Stamminger, Real-time 3D reconstruction at scale using voxel hashing, ACM Trans. Graphics (TOG) 32 (6) (2013) 169.
[44] T. Ringbeck, T. Möller, B. Hagebeuker, Multidimensional measurement by using 3-D PMD sensors, Adv. Radio Sci. 5 (7) (2007) 135–146.
[45] A. Sabov, J. Krüger, Identification and correction of flying pixels in range camera data, in: Proceedings of the Spring Conference on Computer Graphics, 2010, pp. 135–142.
[46] M. Schmidt, B. Jahne, Efficient and robust reduction of motion artifacts for 3D time-of-flight cameras, in: Proceedings of the International Conference on 3D Imaging (IC3D), 2011, pp. 1–8, doi:10.1109/IC3D.2011.6584391.
[47] J. Smisek, M. Jancosek, T. Pajdla, 3D with kinect, in: IEEE International Conference on Consumer Depth Cameras for Computer Vision (CDC4CV), 2011, pp. 1154–1160.
[48] T. Stoyanov, A. Louloudi, H. Andreasson, A.J. Lilienthal, Comparative evaluation of range sensor accuracy in indoor environments, in: European Conference on Mobile Robots (ECMR), 2011, pp. 19–24.
[49] T. Stoyanov, R. Mojtahedzadeh, H. Andreasson, A.J. Lilienthal, Comparative evaluation of range sensor accuracy for indoor mobile robotics and automated logistics applications, Rob. Auton. Syst. 61 (10) (2013) 1094–1105.
[50] L. Vera, J. Gimeno, I. Coma, M. Fernández, Augmented mirror: interactive augmented reality system based on kinect, in: Human-Computer Interaction–INTERACT, in: LNCS, 6949, Springer, 2011, pp. 483–486.
[51] M. Wiedemann, M. Sauer, F. Driewer, K. Schilling, Analysis and characterization of the PMD camera for application in mobile robotics, in: Proceedings of the IFAC World Congress, 2008, pp. 6–11.
[52] Z. Xu, T. Perry, G. Hills, Method and system for multi-phase dynamic calibration of three-dimensional (3D) sensors in a time-of-flight system, US Patent 8,587,771, 2013.
[53] Z. Xu, R. Schwarte, H. Heinol, B. Buxbaum, T. Ringbeck, Smart pixel – photonic mixer device (PMD), in: Proceedings of the International Conference on Mechatronics & Machine Vision, 1998, pp. 259–264.
[54] C. Zach, T. Pock, H. Bischof, A duality based approach for realtime TV-L 1 optical flow, in: Proceedings of German Conference on Pattern Recognition (DAGM), Springer, 2007, pp. 214–223.
[55] L. Zhang, B. Curless, S.M. Seitz, Rapid shape acquisition using color structured light and multi-pass dynamic programming, in: IEEE International Symposium on 3D Data Processing, Visualization, and Transmission, 2002, pp. 24–36.
[56] S. Zhang, P. Huang, High-resolution, real-time 3D shape acquisition, in: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW), 3, IEEE Computer Society, Washington, DC, USA, 2004, pp. 28–36.
[57] Z. Zhang, A flexible new technique for camera calibration, IEEE Trans. Pattern Anal. Mach. Intell. 22 (11) (2000) 1330–1334.