

Learning Depth Calibration of Time-of-Flight Cameras

David Ferstl

ferstl@icg.tugraz.at

Christian Reinbacher

reinbacher@icg.tugraz.at

Gernot Riegler

reinbacher@icg.tugraz.at

Matthias R  ther

ruether@icg.tugraz.at

Horst Bischof

bischof@icg.tugraz.at

Graz University of Technology

Institute for Computer Graphics and
Vision

Inffeldgasse 16, 8010 Graz, AUSTRIA

Abstract

We present a novel method for an automatic calibration of modern consumer Time-of-Flight (ToF) cameras. Usually, these sensors come equipped with an integrated color camera. Albeit they deliver acquisitions at high frame rates they usually suffer from incorrect calibration and low accuracy due to multiple error sources. Using information from both cameras together with a simple planar target, we will show how to accurately calibrate both color and depth camera, and tackle most error sources inherent to ToF technology in a unified calibration framework. Automatic feature detection minimizes user interaction during calibration. We utilize a Random Regression Forest to optimize the manufacturer supplied depth measurements. We show the improvements to commonly used depth calibration methods in a qualitative and quantitative evaluation on multiple scenes acquired by an accurate reference system for the application of dense 3D reconstruction.

1 Introduction

Time-of-Flight (ToF) sensors are widely used in many applications such as autonomous navigation, 3D reconstruction and human-computer-interaction. Thanks to an adoption by the gaming industry, ToF cameras have become reliable and affordable and are present in many living rooms as an accessory to gaming consoles. These modern devices combine a depth sensor with a traditional RGB sensor to form a so-called RGB-D camera.

Unlike passive 3D sensors based on stereo triangulation, ToF cameras emit light and therefore deliver acquisitions mostly independent from lighting conditions resulting in dense, real-time depth-maps of nearly arbitrary surfaces. The quality of the depth-maps depends on the scene geometry and surface properties in the scene. The two main error sources are 1) intrinsic calibration errors and 2) a spatially varying, context sensitive error in measured depth.

This paper tackles both error sources of a RGB-D camera system using a novel calibration framework. The main goal of this work is to provide a highly accurate calibration with a minimum of user interaction and without relying on a complex, specifically machined 2.5D calibration target as commonly used in literature [1]. The developed method is generally applicable to depth sensors which also deliver an intensity image and are rigidly connected to a RGB camera. Popular examples of these devices are Intel Sens3D, Microsoft Kinect for Windows v2 (K4Wv2) or the upcoming Google Tango mobile device.

In our method the features are automatically detected on a calibration target with sub-pixel accuracy which are used in both single-camera and stereo calibration. We treat the intensity image of the depth camera as a regular intensity camera which allows us to parameterize the depth camera using a standard pinhole model with non-linear lens distortion. After estimating the extrinsic and intrinsic parameters of depth and RGB camera we aim at minimizing the spatially varying depth bias which is inherently present in ToF-based depth cameras [2]. It has been shown that this error bias depends on scene geometry, reflection properties of the scene and position of the pixel in image space [3, 4]. Instead of directly modeling each error distribution with heuristics, we solve the problem of depth-bias calibration directly by using machine learning. In our framework we use a Random Regression Forest (RRF) to directly infer the mapping from depth and intensity features to a depth offset, which jointly eliminates all error sources. An overview of our method is shown in Fig. 1(a). We show that our method outperforms existing correction techniques and improves the quality of dense 3D reconstruction.

2 Related Work

The ToF camera is an active sensor that measures the "time of flight" of near-infrared light (NIR) emitted by an ideally coaxial light source. The intensity of the illumination is modulated to measure the depth from the phase shift between emitted and reflected light acquired by the sensor [5]. This principle delivers depth measurements invariant to scene illumination at frame rates up to 160 fps. In the following we will discuss the drawbacks which arise from the projective camera model as well as the active measuring principle.

The ToF camera follows the pinhole camera model which can be parametrized by intrinsic parameters like focal length, skew and principal point. Most ToF cameras have a wide field of view (FOV) which adds severe non-linear lens distortion. All these camera parameters are determined during camera calibration (*i.e.* with the approach of Zhang [6]). This homography-based camera calibration was realized in a Matlab calibration toolbox [7] and uses a planar checkerboard pattern to find the 2D-3D correspondences. Considering the limited resolution of ToF cameras, Kahlmann *et al.* [8] use a slightly modified pattern for this calibration.

Since the amplitude modulation on the emitting light source of ToF cameras is not optimally sinusoidal, a periodic distance related offset occurs, namely the *wiggling error*. Fuchs and Hirzinger [9] model this error as a third order polynomial. Additionally, due to different reflection properties an *intensity related error* exists. On the one side low intensity measurements lead to a bad Signal to Noise Ratio (SNR) and hence to *random noise*. On the other side there is a relation between reflection properties of a scene and its measured depth. This leads to a detectable offset e.g. lower reflection leads to a depth measurement closer to the camera. An intensity based compensation of this error source has been shown by Lindner and Kolb [10] and Radmer *et al.* [11].

Belhedi *et al.* [2] propose a non-parametric method to compensate for the depth errors. In contrast to previously presented methods no underlying model is assumed, but the measurement volume in front of the camera is discretized and a depth offset is estimated for each voxel in the calibration step. Since not all voxels can be hit during calibration, a regularization term fills in the missing values. The authors only utilize the depth as input feature, in contrast our method also makes use of the intensity image and does not require a volumetric representation and thus no discretization of the world.

Reynolds *et al.* [15] proposed a Random Forest (RF) to quantify the confidence of a ToF measurement. This RF is trained using ground truth depth together with acquired depth and intensity features. As output it delivers a per-pixel-confidence according to each measurement. In spirit, this approach is the closest to the proposed method. In contrast to predicting the measurement confidence, our method goes one step further and directly corrects the depth error.

Discussion During the acquisition process of ToF cameras, measurement errors originate from many different sources. By now, there exist a variety of different calibration techniques to estimate and calibrate each of those errors separately. Unfortunately the different errors highly correlate which naturally makes it hard to calibrate each error on its own. In our work we propose a method to measure and calibrate all of those errors automatically without user interaction and the need of an expensive reference system.

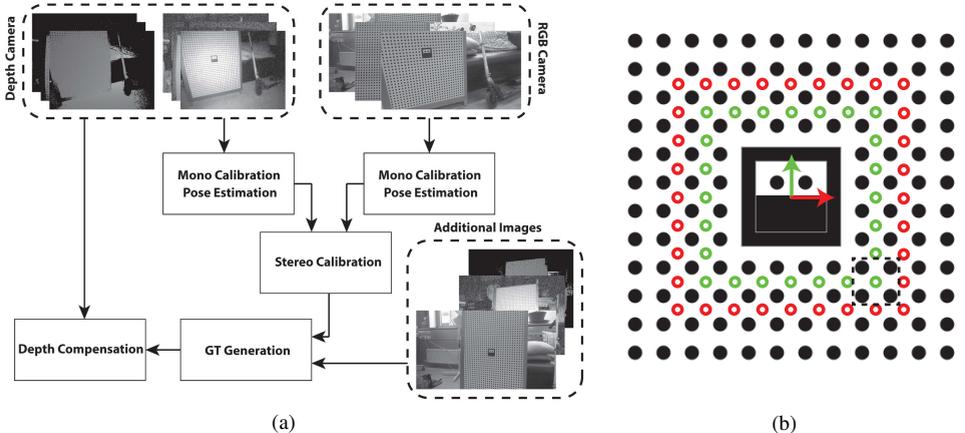


Figure 1: System overview (a). Using a planar target with known dimensions (shown in (b)) the system automatically detects feature points in both the RGB and the depth image. These points are used for a full stereo calibration and serve as ground truth 3D points for depth calibration. Please note that the green and red circles are not part of the printed target, but depict the location of one feature.

3 Method Overview

The goal of depth acquisition is to directly measure dense, metrically correct points in a scene, parametrized in the form of a depth-map \mathbf{D} : $\Omega \subset \mathbb{R}^2$ from a camera at $[0, 0, 0, 1]^T$. Hence, the depth-map consists of pixel-wise depth measurements $\mathbf{D}(\mathbf{x})$ at $\mathbf{x} = [i, j, 1]^T$ over

the whole image space Ω . Each depth measurement can be projected to world space by

$$\mathbf{X} = \mathbf{D}(\mathbf{x}) \frac{\mathbf{K}^{-1}\mathbf{x}}{z} \quad (1)$$

$$z = \begin{cases} \|\mathbf{K}^{-1}\mathbf{x}\| & \text{for a perspective projection} \\ 1 & \text{for an orthographic projection} \end{cases} \quad (2)$$

where \mathbf{K}^{-1} is the inverse of the intrinsic camera matrix. The reprojection in Eqn. (1) can be inaccurate due to two error sources which limit the depth measurement performance: 1) an error in intrinsic camera calibration which affects the direction of the rays encoded in \mathbf{K} and 2) a context sensitive depth bias encoded in $\mathbf{D}(\mathbf{x})$. In the following sections we will show how we can tackle both error sources. An overview of the framework is shown in Fig. 1(a). In Section 4 we present our calibration framework where we introduce a method to automatically detect the features on a circular pattern target. The target is especially suited for cameras with rather low resolution, like in modern depth sensors. The detected features are not only used to determine intrinsic and extrinsic parameters of RGB and depth cameras but also serve as Ground-Truth (GT) to our depth compensation method presented in Section 5. We treat the depth error correction as a regression problem and learn a direct mapping between the acquired depth and intensity to the offset from the real depth using a Random Regression Forest (RRF).

4 Geometric Camera Calibration

Geometric camera calibration serves the purpose of determining the intrinsic and extrinsic parameters of depth and RGB camera. Since the main focus of this work is to provide a user-friendly and fully automatic calibration pipeline for both geometric and depth camera calibration, we utilize a proven camera calibration method of [9] in conjunction with a circular pattern target and automatic feature detection presented in the following.

Calibration Target Traditional camera calibration methods utilize a checkerboard target with known dimensions to establish a correspondence between 2D features and 3D world points. The accurate detection of checkerboard crossings or circular features is very inaccurate for low-resolution cameras. Methods for calibrating depth cameras therefore usually utilize 2.5D targets which have to be custom-built. In this work we aim to use a conventional 2D target and define feature points by a *combination* of multiple circular shapes printed on a planar surface (depicted as dashed rectangle in Fig. 1(b)). Fig. 1(b) shows a cut-out of the proposed calibration target, overlaid with the coordinate system origin and the feature point positions (in red and green). The actual dimensions of the target are flexible and can be adapted to the application at hand. The target feature points $\mathbf{X}_i \in \Omega_T$ are defined on a regular grid relative to the coordinate origin in 3D world coordinates.

Feature Extraction Our automatic feature extraction method consists of two parts: 1) detecting the center marker and 2) iteratively detecting the circular targets, similar to the concept of [10]. This two stage approach has the advantage that the whole target does not have to be visible in all images, a drawback of many other calibration frameworks. The detection of the central marker is done using the marker detection of [9] which results in the

four corner points of the marker in image space. We estimate a homography between the detections and the known positions in target space which allows us to transform all features from target space to image space and determine the feature positions. The features are then detected by correlating the image region around the estimated point positions with a known template, warped with the estimated homography. In practice non-linear distortions in image space make this simple approach not applicable. However, for a small image patch the assumption of a local homography is reasonable. We therefore start the detection with feature points in the vicinity of the center marker (depicted in green in Fig. 1(b)). After successful detection, the search region is enlarged and the detected feature points in the neighborhood around a new feature point are used to compute a new local homography. The estimated feature location is again refined using template matching. This procedure is iterated until either the borders of the target are reached or the correlation score drops below a threshold. The feature positions are locally refined with a few iterations of the Baker-Matthews Inverse Compositional Algorithm [10].

The output of the feature detection is a set of correspondences between the input image and known 3D positions on our calibration target¹ which can be used in conjunction with the MATLAB toolbox of [9]. Example detection results on challenging images can be seen in Fig. 2. With the knowledge about camera parameters at hand we are able to correct the ray direction of Eqn. (1) with the updated values for \mathbf{K} . In the next section we will tackle the second error source, the context sensitive depth offset.

5 Depth Error Compensation

Time-of-Flight (ToF) cameras suffer from errors like wiggling error and reflectivity dependent error as shown in detail in Section 2. While being systematic, they depend on a variety of parameters which simultaneously influence the measurement such as scene geometry, surface reflectivity, distance and orientation of the camera with respect to the scene.

Instead of explicitly modeling the depth error for the various error sources, we formulate the non-linear mapping between input depth and intensity features to a depth offset as a regression problem. We propose to use a modified Random Forest (RF) in a regression setting [11] to directly calculate the offset between measured depth and GT depth. To train the RF we have given a sample set $\mathcal{X} = \{\mathbf{D}_k, \mathbf{I}_k\}_{k=1}^K$ consisting of K depth and intensity images from an intrinsically calibrated depth camera together with corresponding target set given by $\mathcal{Y} = \{\mathbf{D}_k - \mathbf{D}_k^{\text{GT}}\}_{k=1}^K$. According to the general regression formulation, the training set is given by $\{\theta(\mathcal{X}, \mathbf{x}_i), y(\mathcal{Y}, \mathbf{x}_i)\}_{i=1}^N$, where $\theta(\mathcal{X}, \mathbf{x}_i) \in \mathbb{R}^M$ and $y(\mathcal{Y}, \mathbf{x}_i) \in \mathbb{R}$. In this formulation $\theta(\mathcal{X}, \mathbf{x}_i)$ is a M dimensional feature vector extracted from the depth and intensity images and $y(\mathcal{Y}, \mathbf{x}_i)$ is the offset from input to GT depth. The training set is extracted at N pixel positions \mathbf{x}_i which are randomly sampled out of \mathcal{X} and \mathcal{Y} .

The trained RF in our setting describes the non-linear mapping $\mathcal{M}: \mathbb{R}^M \rightarrow \mathbb{R}$, where a new feature vector θ is mapped to a predicted offset y . This mapping is learned by an ensemble of binary decision trees $\{\mathcal{T}_i\}_{i=1}^T$ (T being the number of trees), each trained on a subset of the training data. A single decision tree \mathcal{T}_i recursively splits the given training data into two partitions, such that the uncertainty of the target variables in the resulting subset is minimized. In particular, each node in a tree randomly samples a set of splitting functions,

¹The MATLAB code for the automatic feature detection is available at <http://rvlab.icg.tugraz.at/calibration> since we believe that it can be of use for a variety of camera calibration tasks.

each separating the data into two disjoint subsets. All splitting functions are then evaluated according to their information gain (differential entropy). The splitting function giving the highest gain is fixed and the data is separated into the subsets. This procedure continues until the maximum tree depth or the minimum number of samples left is reached, resulting in a density model for each leaf node.

After evaluating the forest for a given sample, the predicted offset y^* is calculated as the median over the predicted target variables. Hence, the forest gets robust against gross outliers which usually can occur in ToF depth measurements.

In the following section we will show how the training data $y(\mathbf{x}_i) = \mathbf{D}^{\text{GT}}(\mathbf{x}_i) - \mathbf{D}(\mathbf{x}_i)$ is generated from the images captured during geometric camera calibration.

Ground Truth Generation Having an accurate ground truth data for learning the depth compensation is essential. Commonly used methods involve the use of an external measurement system (e.g. [14, 15]) or controlled movement of the sensor system (e.g. [4]). In this work we make use of the integrated RGB camera in modern ToF cameras together with the proposed planar target.

For each color image where the target has been detected, correspondences between feature points in image space and their locations on the known calibration target in 3D space are given. These are only sparse detections at the feature locations itself, but we can assume the points in between to lie on the same plane. The points inside the convex hull of all feature point positions in 3D are transformed to the depth camera coordinate system with the known extrinsic calibration. Using the intrinsic camera parameters of the depth sensor \mathbf{K} and the distortion parameters, a viewing ray for each depth measurement is calculated. Instead of undistorting the image, we distort the point locations, circumventing errors due to an interpolation in image space. The viewing rays which lie inside the projected convex hull are intersected with the plane defined from the high-resolution color camera. Hence, we get a pixel-accurate GT which is orders of magnitude more accurate than the depth camera measurement. An example detection result can be seen in Fig. 2, where the detected feature locations in the color image are projected into the coordinate system of the depth camera. The depth measurements inside the green area are used for training data extraction.

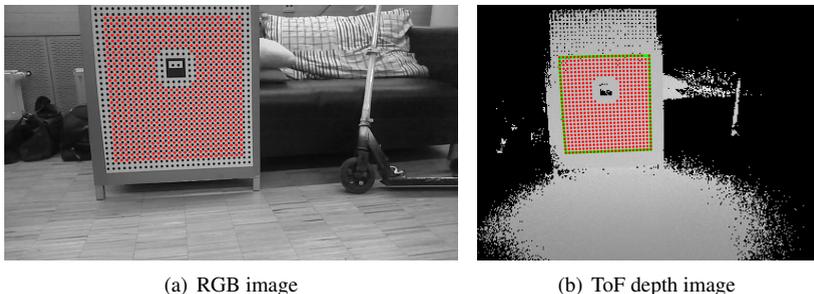


Figure 2: Sample detection result. Feature points are localized in the RGB image (a) and projected into the depth image (b) which has a comparatively lower resolution. The area delineated by green lines is used for training data.

6 Experiments

For the experimental evaluation of our calibration method we use a modern consumer depth camera based on ToF technology, namely the Intel Sens3D. It consists of a depth camera with a resolution of 320×240 pixels and a RGB camera with a resolution of 1280×720 pixels. The recommended depth range for this camera is ≈ 300 to 1000 mm. The physical dimensions of the used calibration target are 600×900 mm, printed on white paper and glued on a glass plate, as shown in Fig. 2. We begin the quantitative evaluation of our calibration framework by analyzing the performance of the individual components. To that end we compare the results of the geometric calibration routine to a known external measurement system. We further evaluate the features we use in our RF. The performance of the depth calibration is compared to the depth without calibration, to depth calibrated using the manufacturer intrinsics and distortion parameters, and to baseline methods which either focus on the distance related error of ToF cameras (known as *wiggling error*) or the *intensity based error*. We evaluate the depth error based on the calibration data as well as on a desktop scene which was acquired by a high accuracy reference system.

6.1 Geometric Calibration

A key component of the presented calibration framework is the automatic feature detection and GT data generation for the subsequent depth error compensation. The accuracy of intrinsic camera calibration and the estimation of the extrinsic camera parameters directly influence the quality of the GT data.

For a geometric camera calibration we use our calibration method based on [9] along with the proposed automatic feature extraction method, described in Section 4. To validate the feature detection we intrinsically calibrate the RGB camera using 20 images of our target. The mean reprojection error of the image features after calibration was estimated to be 0.12px. To translate this value into the metric world, we assess the accuracy of estimated camera poses by rigidly mounting the RGB-D camera to a Leica Disto D8 laser meter². Using this setup we took 20 additional images by moving the camera system coaxial to the optical axis of the RGB camera in the range of 40 to 150cm distance to the target. Keeping the intrinsic parameters fixed, we estimate the extrinsic parameters (*i.e.* the camera pose) for each image. The estimated camera pose enables us to calculate the distance between any point on the calibration target plane to the camera. Additionally, for each image we took a GT measurement using the laser meter. Over the complete range the deviation of the camera pose with respect to the laser measurement is 0.48 ± 0.37 mm.

This experiment shows that the camera pose estimation is at least one order of magnitude more accurate than the typical accuracy specified by the manufacturers of ToF cameras and deems it feasible to be used in our GT data creation detailed in Section 5.

6.2 Features for Depth Calibration

The random forest trained with data from the calibration process solves a regression problem to find a mapping from a feature vector $\theta(\mathcal{X}, \mathbf{x})$ to a depth offset $y(\mathcal{Y}, \mathbf{x})$. The feature vector used throughout our experiments is solely based on features from the intensity and depth

²The accuracy according to the manufacturer is 1/32 inch.

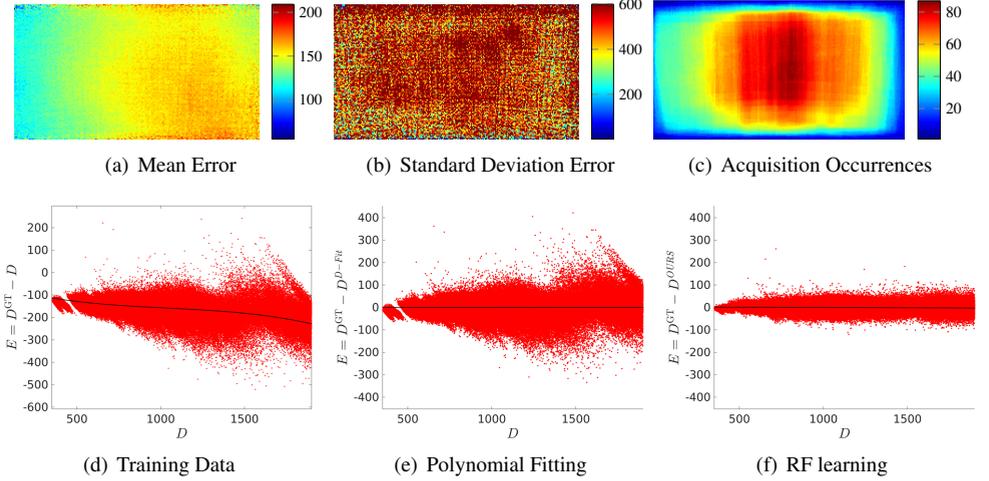


Figure 3: Depth error distribution in training data. The first row shows the error distribution in images space in the captured training set, where (a) mean and (b) standard deviation of the discrepancy between depth camera and ground truth depth are shown. (c) shows the number of times a specific pixel was added to training data. The second row shows the error along the measured depth with the standard camera output (d), after polynomial fitting of the depth (e) and after calibration using our learning based approach (f).

image of the depth camera and is defined as

$$\theta(\mathcal{Y}, \mathbf{x}) = \left[\mathbf{D}(\mathcal{N}_{\mathbf{x}}), \mathbf{I}(\mathcal{N}_{\mathbf{x}}), \text{var}(\mathbf{D}(\mathcal{N}_{\mathbf{x}})), \frac{\partial^2}{\partial x, y} \mathbf{D}(\mathbf{x}), \frac{\partial}{\partial x} \mathbf{D}(\mathbf{x}), \frac{\partial}{\partial y} \mathbf{D}(\mathbf{x}), r(\mathbf{x}) \right], \quad (3)$$

where $\mathcal{N}_{\mathbf{x}}$ represents patches in the $n \times n$ neighborhood around point \mathbf{x} . These features are useful to calibrate depth offsets according to the measured depth (*wiggling error*, *depth error*) and to the measured intensity (*intensity related error*). To further reduce the influence of statistical noise we use the variance and the first and second derivatives of depth as additional features. This is combined with the pixel position encoded as Euclidean distance $r(\mathbf{x})$ to the principal point (*pixel related error*). In Fig. 3 we evaluate the distribution of the error present in the training data. As can be seen, there is a bias towards the edges of the image, in mean and standard deviation. The resulting dimension of our feature vector is $M = 2n^2 + 5$.

Feature Importance During training, the RF selects among the presented features the most relevant to the task at hand. The importance of each feature can be inferred from how often it has been selected. We analyzed the selected features of the proposed random forest for each depth of the tree and visualize it in Fig. 4.

The most important features are derivations of the input depth followed by the position in the image and the raw depth and intensity patches. In the early nodes of the tree, the first and second derivatives are most impor-

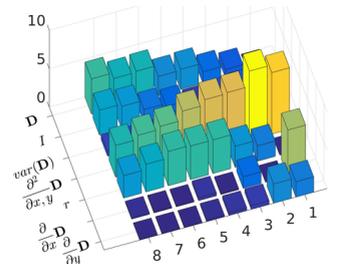


Figure 4: Relative Feature importance of the RF across the forest level

tant, while the raw pixel values become more important for higher tree depths.

In our experiments we use a patch size of $n = 3$. With the defined feature vector the RF is trained using 16 trees with a maximum tree depth of 8. For training we extracted ≈ 2.5 mill. point correspondences from 180 images. The forest is trained using and train-test split of 80/20% of the extracted points.

6.3 Depth Calibration

To evaluate the accuracy of our method we compare the calibration result using the RF to the manufacturer calibration (*S3D-calib*) and to fitting for depth error compensation [6, 6, 7, 24], where a polynomial function is fitted either to the measured depth (*D-Fit*) or the intensity (*I-Fit*) separately. For fitting we deliberately use the same train-test split as for the RF.

The *Test Set* result is visualized in Fig. 3(d-f) and Tab. 1. In this evaluation we show the performance of the depth calibration using our feature selection in an RF compared to polynomial fitting methods. After calibration by polynomial fitting for depth and intensity values, the mean error over the whole training set is reduced to a minimum. However since the errors of the ToF measurements are non linear and highly connected to each other the intensity and pixel related errors can not be removed by a fitting in depth and vice versa. In our approach we take a variety of features into account which combine depth, intensity, variance, gradients and pixel position. This drastically reduces the standard deviation of the error.

We further compare the methods on a real desktop scene. The GT measurement is generated using a Structured Light (SL) scanner which consists of two 4MP intensity cameras and one projector. The depth uncertainty at the given baseline is 1.2 mm. The acquired *Desk Scene* is selected to incorporate a high texture variation. Since the utilized depth camera has a depth range of ≈ 300 to 1000 mm the scene size is chosen appropriately. We compare the different methods both on single ToF acquisitions as well as full 3D reconstructions using the KinectFusion framework of Newcombe *et al.* [13] (named *kFusion* in the following), where a series of 70 depth images is integrated into a volumetric representation. The image sequence is fed into the *kFusion* algorithm before and after the our depth calibration. The results of the fitting compared to our calibration on the *Test Set* as well as the *Desk Scene* are shown in Fig. 5 and Tab. 1.

	<i>Test Set</i>				<i>Desk Scene</i>				<i>Desk Scene kFusion</i>			
	<i>D</i>	<i>I-Fit</i>	<i>D-Fit</i>	<i>OURS</i>	<i>D</i>	<i>S3D-calib</i>	<i>I-Fit</i>	<i>D-Fit</i>	<i>OURS</i>	<i>D</i>	<i>S3D-calib</i>	<i>OURS</i>
RMSE(E)	160.01	40.24	32.03	16.10	168.48	158.23	47.45	46.09	39.83	205.25	167.61	88.60
\bar{E}	-155.51	-0.33	-0.12	-0.02	-141.19	-139.33	-23.40	-18.27	2.65	-187.05	-149.30	-52.18
$\sigma(E)$	37.70	40.24	32.03	16.10	91.92	74.99	41.67	42.32	39.74	84.51	76.179	71.61

Table 1: Accuracy evaluation. The depth error is measured as RMSE(E), mean error \bar{E} and standard deviation $\sigma(E)$ to GT in mm. The accuracy is evaluated for the *Test Set* of the recorded samples from the training data, a recorded *Desk Scene* and the 3D reconstruction of 110 images of the *Desk Scene* using the *kFusion* framework. The error of our method is compared to the uncalibrated camera output (*D*), the camera output calibrated with the values provided by the manufacturer (*S3D-calib*), to error fitting to ToF intensity (*I-Fit*) and to error fitting to depth (*D-Fit*).

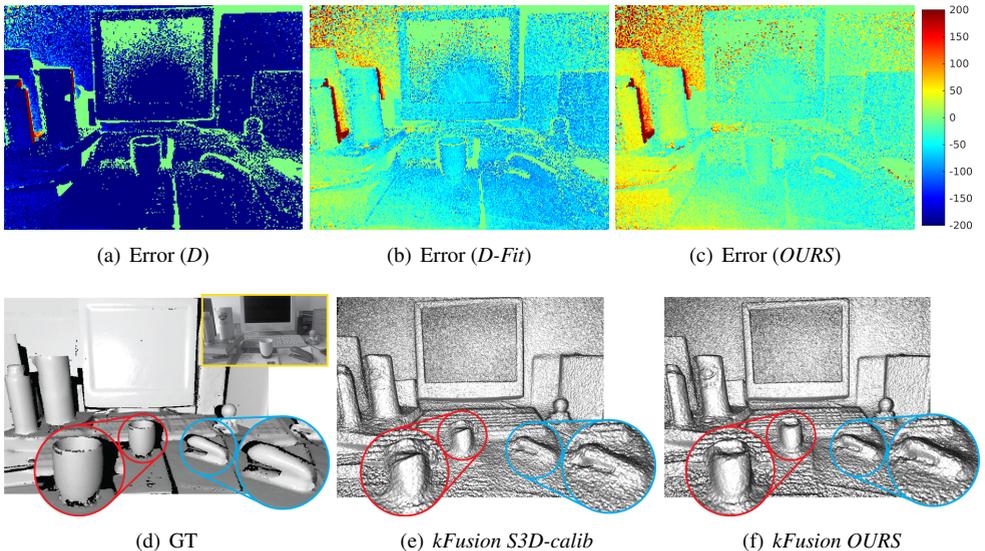


Figure 5: *Desk Scene* Evaluation. The first row shows the color-coded error to GT(d) for one depth acquisition. It compares the uncalibrated camera output D (a), the camera output corrected with depth dependent polynomial fitting D -Fit (b) to our method $OURS$ (c). Regions where no depth data is available are marked with zero. The second row shows the rendered $kFusion$ output of an image series of the *Desk Scene* with magnified regions of interest.

7 Conclusion

We proposed a method for the fully automatic calibration of consumer ToF RGB-D cameras. In a first step the intrinsic parameters of both the RGB and the depth camera are calibrated and the relative camera pose is estimated. For this calibration we used a novel calibration target, where the feature points are automatically detected at sub-pixel accuracy. In the second step the errors of ToF depth measurements are calibrated. Since there exist a variety of error sources which are highly connected and can not be estimated by simple functions alone, we utilize an RF in a regression setting to optimize for all error sources and their connections at once. In our evaluations we show that our calibration method delivers highly accurate calibration results compared to the manufacturer settings and to standard fitting techniques as used in other works. We additionally show the improved quality for dense 3D reconstruction.

Acknowledgments

This work was supported by *Infineon Technologies Austria AG*, the Austrian Research Promotion Agency (FFG) under the *FIT-IT Bridge* program, project #838513 (TOFUSION), the research initiative *Mobile Vision* with funding from the AIT and the Austrian Federal Ministry of Science, Research and Economy HRSM programme (BGBl. II Nr. 292/2012).

References

- [1] S. Baker and I. Matthews. Equivalence and efficiency of image alignment algorithms. In *CVPR*, 2001.
- [2] A. Belhedi, S. Bourgeois, V. Gay-Bellile, P. Sayd, A. Bartoli, and K. Hamrouni. Non-parametric depth calibration of a tof camera. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 549–552, Sept 2012. doi: 10.1109/ICIP.2012.6466918.
- [3] J.-Y. Bouguet. Camera calibration toolbox for matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/, 2013. Accessed May, 2015.
- [4] L. Breiman. Random forests. *ML*, 45(1):5–32, 2001.
- [5] S. Fuchs and G. Hirzinger. Extrinsic and depth calibration of tof-cameras. In *CVPR*, 2008.
- [6] S. Fuchs and S. May. Calibration and registration for precise surface reconstruction with time-of-flight cameras. In *Int. Journal of Intelligent Systems, Technologies and Applications*, volume 5, pages 274–284, 2007.
- [7] J. Jung, J.-Y. Lee, Y. Jeong, and I.S. Kweon. Time-of-flight sensor calibration for a color and depth camera pair. *TPAMI*, PP(99):1–1, 2014.
- [8] T. Kahlmann, F. Remondino, and H. Ingersand. Calibration for increased accuracy of the range imaging camera swissranger. In *Symposium on Image Engineering and Vision Metrology.*, 2006.
- [9] H. Kato and M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *IWAR*, 1999.
- [10] Y. M. Kim, D. Chan, C. Theobalt, and S. Thrun. Design and calibration of a multi-view tof sensor fusion system. In *CVPR Workshops*, 2008.
- [11] R. Lange. *3D Time-of-Flight distance measurement with custom solid-state image sensors in CMOS/CCD technology*. PhD thesis, Department of Electrical Engineering and Computer Science at University of Siegen, 2000.
- [12] M. Lindner and A. Kolb. Calibration of the intensity-related distance error of the pmd tof-camera. In *Intelligent Robots and Computer Vision*, 2007.
- [13] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *ISMAR*, 2011.
- [14] J. Radmer, P.M. Fuste, H. Schmidt, and J. Kruger. Incident light related distance error study and calibration of the pmd-range imaging camera. In *CVPR Workshops*, 2008.
- [15] M. Reynolds, J. Dobos, L. Peel, T. Weyrich, and G.J. Brostow. Capturing time-of-flight data with confidence. In *CVPR*, 2011.

- [16] C. Wengert, M. Reeff, P.C. Cattin, and G. Szekely. Fully automatic endoscope calibration for intraoperative use. In *Bildverarbeitung fuer die Medizin 2006*, pages 419–423. 2006.
- [17] Z. Zhang. A flexible new technique for camera calibration. *TPAMI*, 22(11):1330–1334, 2000.