Pacific Graphics 2011
Bing-Yu Chen, Jan Kautz, Tong-Yee Lee, and Ming C. Lin
(Guest Editors)

Volume 30 (2011), Number 7

Exposure Fusion for Time-Of-Flight Imaging

Uwe Hahne[1] and Marc Alexa[1]

[1]TU Berlin, Germany

Abstract

This work deals with the problem of automatically choosing the correct exposure (or integration) time for time-of-flight depth image capturing. We apply methods known from high dynamic range imaging to combine depth images taken with differing integration times in order to produce high quality depth maps. We evaluate the quality of these depth maps by comparing the performance in reconstruction of planar textured patches and in the 3D reconstruction of an indoor scene. Our solution is fast enough to capture the images at interactive frame rates and also flexible to deal with any amount of exposures.

Categories and Subject Descriptors (according to ACM CCS): IMAGE PROCESSING AND COMPUTER VISION [I.4.8]: Scene Analysis—Range data COMPUTER GRAPHICS [I.3.3]: Picture/Image Generation—Bitmap and framebuffer operations

1. Introduction

Time-of-flight depth sensing is utilized in more and more fields of application, among others simultaneous localization and mapping (SLAM), motion capturing in gaming and pedestrian detection in automobiles [MWSP06, YIM07, OBL*05, KBKL09]. All of these applications need the sensor to capture reliable depth data within a range of several meters.

Many time-of-flight cameras work after the following principle: a light source emits an amplitude modulated sinusoidal signal. This signal is reflected by the target and captured by a CMOS based chip. The chip is synchronized with the light source and takes several exposures per cycle of the emitted signal. From these samples the phase shift can be calculated. This phase shift is directly proportional to the distance between sensor and target.

The integration time indicates how long the chip is exposed before the samples are integrated in order to calculate the desired phase shift. A deficiency in numbers of samples leads to uncertain results dominated by noise. On the other hand, too many samples potentially lead to saturation and photons are no longer counted, which also results in errors. Hence setting the correct integration time is crucial for correctly measuring the distance. Usually, the operator of the device has to define the integration time for the sensor manually. May et al. [MWSP06] have proposed a control mechanism that adjusts the integration time during operation. Here, the integration time is set by means of a feedback controller that assumes that the integration time is optimal when the mean intensity of the captured image is at a pre-defined ideal value.

Nevertheless, such an auto-exposure mode is only able to find one globally optimal exposure time for one scene. And just as in regular photography, the resulting image might still have under- and over-exposed regions. As the depth measurement relies on the reflection of an emitted light signal, under- and over-exposed regions lead to errors in depth estimation, as explained above. In fact, each pixel has its own optimal integration or exposure time, which *unlike traditional photography* depends on the *distance and reflectivity* of the captured object itself.

In this paper, we adapt recent solutions from computational photography to this problem. Instead of optimizing for a global integration time, we capture several images and search locally in each exposure for regions which provide most accurate distance data. This poses two challenges: first, the sources of error in the sensor are different from traditional over- and under-exposure in imaging. Second, time-of-flight depth sensing is useful mostly in real time applications meaning the solution has to be computed in fractions of a second.

We implemented a method for capturing time-of-flight range maps in order to provide high quality depth data for the full theoretic range of the camera. Our work is in-

spired by high dynamic range (HDR) imaging, but faces the challenge of dealing with depth instead of color information. Therefore, we propose new measures for the quality of the depth data locally in the original images that lead an image fusion process. These measures are both inspired by a similar approach for color images by Mertens et al. [MKV07] and founded on research in image quality measures [MB09, Gos05]. The time-of-flight camera we use returns amplitude images that refer to the amount of light that has been captured by the sensor. We use only these images and the distance data to compute our quality measures. Hence, our solution does not need any calibration process in order to enhance the depth images. We implemented a real time solution that exploits the capability of the PMD[vision]®CamCube 3.0 camera to capture four images with varying integration times almost at once.

In order to demonstrate the superior quality of the fused depth maps, we captured known planar objects at varying depths and measured the error from the residuals of a least-square plane fitting in the planar regions of the image. Our results show that the fused data is more accurate than data from the ideal exposure time even for a single planar depth region and gives much lower errors if several planar regions at varying depths are taken into account.

We further apply our fused range maps for point cloud alignment and compare the results of the 3D reconstruction of indoor environments with them produced by single exposure depth images.

To our knowledge, we present in this paper the first approach to combine several exposures with varying integration times of a time-of-flight camera in order to enhance the quality of the depth maps. In the following section we relate our work to approaches concerning the integration time in time-of-flight imaging as well as alternative approaches to enhance the dynamic range of the depth sensing.

2. Related Work

First, the camera manufacturers try to enhance the dynamic range as much as possible. MESA Imaging, producer of the Swissranger™ cameras has developed a solution that allows to control the integration time per pixel individually [BOL*05]. Such an enhanced pixel stops the integration as soon as the capacitance exceeds a pre-defined threshold. Unfortunately, the power consumption of such a pixel enhancement is too high in practice and additionally, the used integration time for each pixel has to be stored and transfered in order to reconstruct a homogeneous intensity image. This lead MESA Imaging to not implement such a feature in their products.

There are ambitions to extend the range for time-of-flight imaging by PMDTec as they are offering a plugin to enable modulation frequencies down to 1 MHz. This enhances the working range theoretically up to 150 meter. However in practice, the accuracy would be strongly reduced and the illumination unit has to be amplified as well. Nevertheless PMDTec promises an extended range of 30 meter.

There are numerous approaches that deal with denoising the distance data captured by time-of-flight sensors. While most of these approaches either use additional sensors [HA09, HSJS10, LLK08, ZWYD08] or rely on elaborately generated calibration data [KRI06, LK07, LSKK10], our approach can be applied to all time-of-flight cameras that deliver amplitude and distance data without any preparation of the sensor. While receiving very good results, Lindner and Kolb [LK07] use an additional camera and pre-captured calibration data in order to correct the error resulting from differently reflecting objects.

Similar to our approach, Schuon et al. [STDT08] presented a method based on super resolution. Here, several noisy depth maps captured from slightly different positions are combined to one high resolution, high quality depth map. While this approach has been successfully extended and applied to 3D shape scanning [STDT09, CSC*10], it does not provide - in contrast to our approach - the enhanced depth imaging in real-time.

As already mentioned, our approach is inspired by HDR imaging. Here, several exposures of the same scene are captured with varying exposure times. This leads to images with a varying amount of details in different regions of the image. These images are fused together in order to keep the details visible in all image regions. We refer to the book of Reinhard et al. [RWPD05] for a complete overview in HDR imaging.

Usually, the different images are aligned to one HDR radiance map which can not be displayed without specialized devices [DM08]. It has to be transformed back to low dynamic range by tone mapping to enable the visualization on a regular display. This process has been shortened by Mertens et al. [MKV07]. Thereby, the images are fused directly into a single low dynamic range image that contains all the details from a collection of differently exposed images. For each image pixel a weight is calculated and the final result is an affine combination of the images.

The fusion of color images has already been realized by Goshtasby [Gos05]. He proposes a measure for the entropy of each image pixel and fuses the images based on this measure in a gradient-ascent approach which is not capable for real-time applications. In contrast to this, Mertens et al. [MKV07] aim on the same outcome of fusing images with varying exposure times. They propose three quality measures and merge the images in a fast pyramid based algorithm.

These quality measures are not suitable for range maps in general. We therefore adapt the quality measures to the characteristics of time-of-flight range images. The measure should define a confidence value of the depth data as applied in many other approaches [MDH*08, HA09, KBKL09].

While Frank et al. [FPH09] show that the amplitude value is an optimal indicator for the confidence of the range data, Reynolds et al. [RDP*11] demonstrate that a trained random forest outperforms simple amplitude based thresholding mechanisms. Apart from that Foix et al. [FAACT10] recently presented an approach that models the uncertainty only from the depth data. Based on these inconsistencies in the literature we develop and evaluate new measures. We explain our choices in the next section.

3. Algorithm

Our fusion algorithm is similar to the one described by Mertens et al. [MKV07]. We take several exposures of a scene and fuse the depth images together. Each exposure is multiplied per pixel with a weight map W_k where $k = 1...N$ indicates the number of exposures. This weight map is constructed as an affine combination of several individual quality measures. We define

$$W_k = M_C^{w_C} \times M_W^{w_W} \times M_S^{w_S} \times M_E^{w_E}$$

with the quality measures M (resp. **C**ontrast, **W**ell-exposedness, **S**urface and **E**ntropy) and $\times$ denotes an per pixel multiplication. Each quality measure M is weighted with an corresponding exponent $w \in \{0,1\}$. The weight map is normalized so that the weight of all exposures k sums up to one for each pixel.

The fusion is realized as a multiresolution blending. Instead of fusing directly the full resolution depth maps, image pyramids are computed and fused as proposed by Burt and Adelson [BA83]. The resulting fused depth map R can be reconstructed from the Laplacian pyramid $\mathbf{L}\{R\}$. The l-th level is defined by

$$\mathbf{L}\{R\}^l = \sum_{k=0}^{N} \mathbf{G}\{W_k\}^l \times \mathbf{L}\{D_k\}^l,$$

where D_k denotes the depth map from the k-th exposure. Each level l of the pyramid is constructed by a weighted sum of the corresponding levels of a Laplacian pyramid over all exposures. The weights are obtained from the l-th level of the Gaussian pyramid of the weight maps. See Figure 1 for a schematic overview about the process. Note that this fusion scheme slightly enhances the quality, however it can also be replaced by a simpler full resolution blending.

3.1. Quality measures

In this section, we describe the definition of new quality measures for the depth map fusion in detail. Note that these measures are not entirely calculated from the depth images, but also based on the amplitude images. We enumerate the image pixel indices as i and j. The distance image D with distance values D_{ij} is normalized to $[0,1]$ by setting a linear mapping. Distances with the theoretical maximal distance of $7.5m$ are mapped to 1, while a distance of $0m$ is mapped to zero. The amplitude image A with amplitudes A_{ij} is also normalized. Note that the amplitude does not have a theoretical maximum. It is bounded by the technical properties of the chip, hence we bound the maximum at a value where the sensor is not yet saturated.

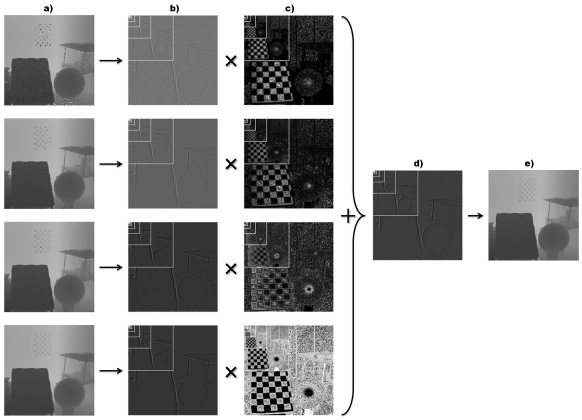

Figure 1: *Exposure fusion principle: a) Captured depth maps, b) Depth map – Laplacian pyramids, c) Weight map – Gaussian pyramid, d) Fused pyramid, e) Final depth map (after [MKV07])*

3.1.1. Contrast M_C

One big issue with time-of-flight depth images are so called *flying pixels*. Due to aliasing effects, the distance along depth discontinuities is computed from photons collected by the sensor from foreground and background. This leads to wrong distances that lie between the values of fore- and background. As the amplitudes along the depth discontinuities are also measured per pixel and hence between the fore- and background values, we can define a quality measure that fosters image regions where the depth discontinuities do not lead to flying pixels. These regions are identified by the contrast in the amplitude image which leads us to define

$$M_C = \|\Delta A\|.$$

We apply a 3x3 Laplacian filter to the amplitudes images and use the absolute values of the filter response which yields an indicator for contrast. In the amplitude images a strong contrast occurs usually along depth discontinuities as the reflectance of the foreground object differs from the object behind. Note that this measure has also been used by Mertens et al. [MKV07] in order to enhance the contrast in the resulting image.

3.1.2. Well-exposedness M_W

For time-of-flight cameras the amplitude image indicates under- or overexposure, hence the amplitude can be used as a confidence measure. As already mentioned, we normalize the amplitude images. We determine amplitude values A_{min}

and A_{max} for under- and overexposure and map all the values in between linearly to the interval $[0,1]$. All values outside this range are mapped to zero or one respectively. We calculate each pixel W_{ij} of this quality measure M_W as

$$W_{ij} = e^{\frac{-(A_{ij}-\alpha)^2}{2\sigma^2}}$$

with $\alpha = 0.5$ and $\sigma = 0.2$. We adapt this quality measure from the so-called well-exposedness measure from Mertens et al. [MKV07]. They argue that intensities close to zero indicate underexposure and close to one overexposure respectively. In our adaption the pixels with an optimal normalized amplitude value of 0.5 get the highest weighting. Note that the critical part is the determination of A_{min} and A_{max}. They can either be obtained from the camera manufacturer or by capturing a wall from a fixed distance. Then plot the mean distance and amplitude values while varying integration time from low to high. The mean distance will change drastically as soon as the sensor is under- or overexposed. From this boundaries A_{min} and A_{max} can be determined.

3.1.3. Surface M_S

Beside these two quality measures that already lead to promising results, we defined a further one based on the measure of the structural similarity [WBSS04] and its adaption to range maps [MB09]. A measure for the surface roughness can be defined as

$$M_S = 1 - \frac{(\sigma - \mu^2)}{max(\sigma - \mu^2)}$$

where σ is the Gaussian filtered version of D^2, while μ is a Gaussian filtered version of D. The difference $\sigma - \mu^2$ correlates with the frequencies in the images. This difference is divided by its maximal value and subtracted from one so that the measure M_S is high in smooth regions. The smoothness indicates the absence of noise and leads to the assumption that the depth values are correct.

3.1.4. Entropy M_E

Similar to the approach from the well-exposedness measure M_W, Goshtasby [Gos05] describes the idea for image fusion to strengthen image regions that contain the most information. A measure for the amount of information is entropy. The entropy measure M_E contains an entropy value E_{ij} for each pixel. It is calculated for a local histogram around each pixel of the image as

$$E_{ij} = -\sum p(A_{ij}) \log_2(p(A_{ij}))$$

where p contains the histogram counts from an 9x9 neighborhood of each pixel and each histogram contains 256 bins. The entropy of an image has the property that it only depends on the image histogram. This leads to a disadvantage of the entropy. The image information is defined as uncertainty which is maximal for noisy images.

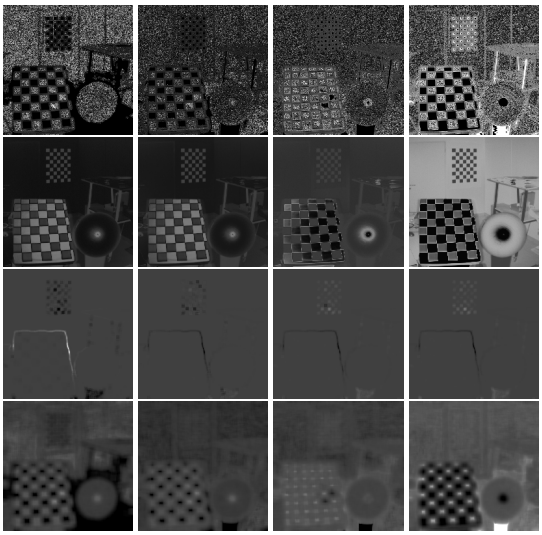

Figure 2: *Comparison of all weights side by side. Each column is one integration time. Rows are sorted from top to bottom showing M_C, M_W, M_S and M_E.*

See Figure 2 for a side by side comparison of all quality measures for an example scene.

3.2. Discussion

We further have to define the number of exposures k that we will use for image fusion. Our algorithm works with an independent number of exposures. In our real-time implementation, we fuse four images because the PMD[vision]® CamCube 3.0 provides a capture mode that allows to take four successive frames without transferring data in between. Due to the short integration times about max. 5 - 10 ms, these four exposures do not differ significantly even for scenes containing motion. Furthermore, the computation costs fusing four images still allow interactive frame rates.

In order to correct the distance values in real-time, the quality measures have to be computed efficiently. We use a profiler for determining the attended computation time of each measure. See Figure 3 for an illustration of the profiling data measured during live capture and display of the fused depth images. The blue bars in the front row display the computation time contribution during the capture, fusion and anything else (e.g. rendering). The red bars are the weighting functions inside the fusion algorithm. This figure clearly shows that the entropy calculation is the bottle neck in our implementation. It leads to a decrease in frame rate in the real-time implementation down to 2-4 fps even if optimized algorithms are used for the calculation of the logarithm [VFM07] and the number of unnecessary additions in the summation is minimized. Without calculating the entropy the whole fusion process is computed in 0.0461 seconds on standard laptop computer with a dual core processor running at 2.16 GHz and equipped with 2 GB of RAM.

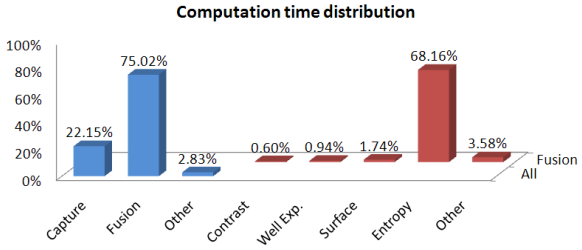

Figure 3: *Results from profiling: blue bars show the distribution for the complete program, red bars the distribution inside teh fusion algorithm.*

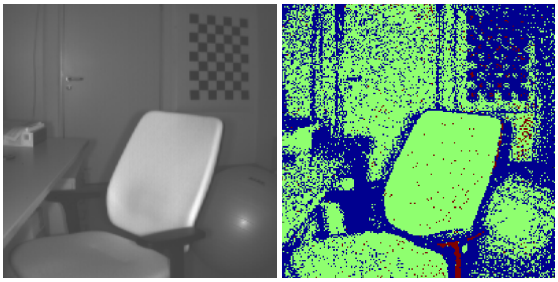

Figure 4: *The static test scene (left: PMD CamCube intensity image, right:comparison of standard deviations single exposure and fused solution)*

We therefore further evaluate our algorithm to identify the impact of each quality measure on the accuracy of the depth maps.

4. Evaluation

We implement several test environments in order to demonstrate the superior quality of the fused depth maps over depth maps acquired with a single integration time.

4.1. Stability over time test

First, we need to define the integration times for our test. Therefore, we implemented a PID controller approach following the ideas of May et al. [MWSP06]. The integration time is set so that the mean intensity of the captured image is optimal. In the following, we refer to this integration time as the ideal integration time t'. In addition, the number of exposures and the integration times for each exposure have to be defined for our test cases. We decided to use following scheme for the first test:

$$t_i = 2^{i-\frac{N}{2}} t'$$

where N is the number of exposures and $i = 0..N$ indicates the exposures.

As a first comparison between the fused images and images captured with the ideal integration time, we analyzed

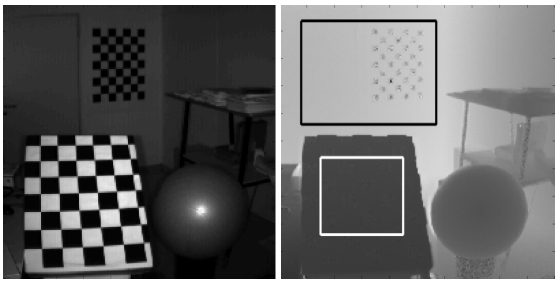

Figure 5: *Plane fitting test scene (left: PMD CamCube intensity image, right: fused depth map with two ROI (white: near, black: far)*

the standard deviation of the distance values of a static scene over a time period of 50 frames. See Figure 4 for an intensity image of the scene (left). We further compare the standard deviations per pixel (right) - we colorize the pixels depending whether the standard deviation is smaller for the single integration time exposures (red) or for the fused solution (blue). All pixels where the standard deviation does not differ more than 5 mm are marked green. This illustrates that the fused values are more stable especially around depth discontinuities and in textured regions. The mean standard deviation over all pixels in the fused images over time is reduced significantly from 2.94 cm to 2.17 cm.

4.2. Plane fitting error

In a second test, we place two planar objects (see the checkerboards in Figure 5) at different depths in the scene. We select these two regions of interest (ROI) manually. Note that the far checkerboard is mounted on the wall. We captured a series of four exposure with the same scheme as in the stability test case.

As our algorithm enhances directly the depth maps, we have to transform the distance values of each pixel in both ROI (near and far) into 3D coordinates. We use the fixed intrinsic parameters of the PMD CamCube to calculate the 3D points for each pixel of our fused depth maps as well as to compute a point cloud from the depth map obtained with by a single exposure.

We then fit a plane into each ROI in 3D coordinates using a principal component analysis. The perpendicular distance from each point to the plane is minimized. The error is defined as this distance. We then compare the mean squared error (MSE) for this plane fit and the computation time over various weighting combinations. For each ROI we compute the MSE for the data captured with the optimal global integration time and for the fusion results using all possible combinations for the weighting exponents. In addition, we measure the computation time for the fusion process (see Table 2). The second rightmost column contains a weighted sum of both MSE. We use the ratio between the MSE from the single exposure as weight for the sum, so that both MSE

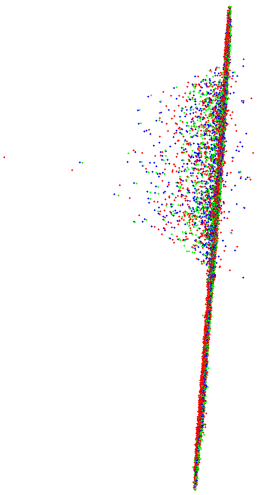

Figure 6: *Close up on the far plane: green dots indicate measures from our fused results, red the single exposure and blue the simple amplitude weighting.*

are equal. The rightmost column shows the error in relation to the weighted sum MSE from the single exposure – values below one indicate an enhancement.

The primary outcome is that the best weighting combination in this setup is a combination of all presented quality measures (see the last row). Our results show that the error is reduced by about nearly 38%.

In order to stress the positive effect of our weights we further compare our solution with a simple approach. We use the normalized amplitude image directly as weight. This results in a small error for the near plane, however the far plane fitting leads to an MSE of 0.02346 what is even larger than in the single exposure. We show the (far) plane fitting results in Figure 6.

We illustrate the correlation of error and computation time in Figure 7. The plot displays the time and error for the combinations of the three most important quality measures - we left out the surface measure M_S for clarity. The contrast measure M_C can clearly be identified as the most effective measure. Adding the well exposedness measure M_W slightly reduces the error. The entropy measure M_E further reduces the error, however at the expense of the computation time. Note that these timings are from the MATLAB implementation, however they confirm the trend from the profiling results of the real-time C++ implementation from Figure 3. Nevertheless the entropy is a suitable quality measure if the computation time is irrelevant.

We did the plane fitting test on further example scenes. Figure 8 shows two walls in an indoor environment. The wall on the right is close to the camera and untextured while the facing wall is textured. Note that the depth variance of the walls is far below the precision of the camera and we can hence assume planarity. We again fit a plane as in the

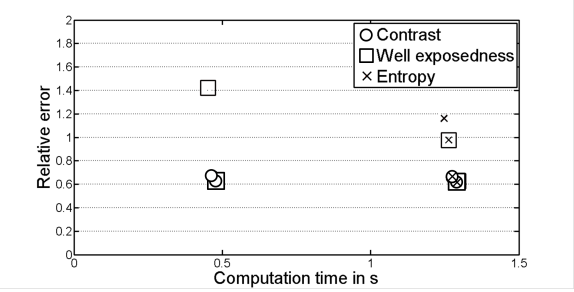

Figure 7: *Plot of error versus timing for various weighting combinations.*

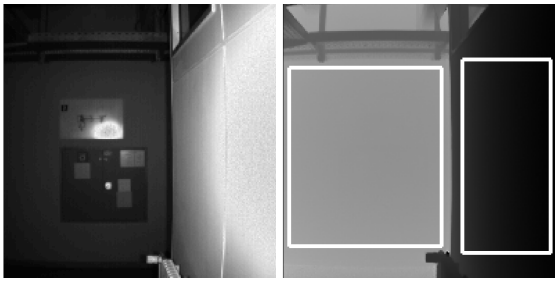

Figure 8: *Second plane fitting test scene (left: PMD CamCube intensity image, right: fused depth map with two ROI (face and right)*

previous example. We compare our fusion result with two simpler approaches. First, we did not apply the multiresolution fusion scheme, but simply computed the *weighted sum* for each pixel. Second, instead of using our derived quality measures, we again use the *normalized amplitude* directly as weight. Beside calculating the MSE, we further compare the estimated angle between the two walls which should be 90° (see Table 1). We included the results from the best single exposure of the sequence.

4.3. 3D reconstruction

An important area for the usage of time-of-flight data are autonomous robots and the SLAM algorithm. In order to determine the position of the robot (and hence the camera) the captured depth maps have to be registered. One way of registering is the alignment of point clouds. We therefore evaluate our method by performing a point cloud alignment by

Method	MSE (face)	MSE (right)	Angle
Multiresolution	**0.07888**	**0.067636**	91.57
Weighted sum	0.07912	0.067819	91.61
Amplitude	0.07895	0.067769	91.60
Single exposure	0.07893	0.067643	**91.48**

Table 1: *Error values from second plane fitting test scene for comparison with simpler weighting and fusion schemes.*

Table 2: *Overview about the effect of each weight on accuracy and computation costs.*

M_C	M_W	M_S	M_E	Timing	MSE (near)	MSE (far)	Weighted sum	Rel. error
0	*0*	*0*	*0*	*0.000*	*0.000761*	*0.0223*	*0.0446*	*1.000*
0	0	0	1	1.248	0.001033	0.0215	0.0518	1.160
0	0	1	0	0.478	0.001689	0.0199	0.0695	1.557
0	0	1	1	1.304	0.001010	0.0168	0.0465	1.041
0	1	0	0	0.452	0.001287	0.0256	0.0634	1.421
0	1	0	1	1.263	0.000786	0.0206	0.0436	0.977
0	1	1	0	0.507	0.001254	0.0190	0.0558	1.251
0	1	1	1	1.317	0.000769	0.0161	0.0386	0.866
1	0	0	0	0.466	0.000414	0.0178	0.0299	0.671
1	0	0	1	1.275	0.000401	0.0175	0.0293	0.657
1	0	1	0	0.504	0.000413	0.0176	0.0297	0.666
1	0	1	1	1.324	0.000400	0.0174	0.0292	0.653
1	1	0	0	0.476	0.000383	0.0171	0.0283	0.634
1	1	0	1	1.296	0.000375	0.0169	0.0279	0.625
1	1	1	0	0.537	0.000383	0.0169	0.0282	0.631
1	**1**	**1**	**1**	**1.358**	**0.000375**	**0.0168**	**0.0278**	**0.623**

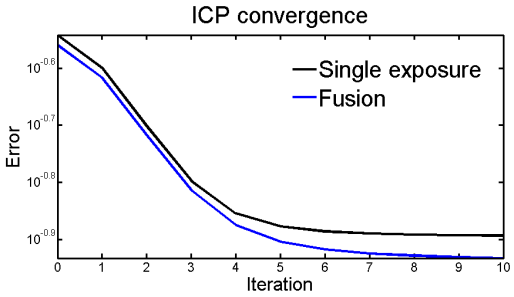

Figure 9: *Convergence of the iterative closest points (ICP) algorithm for single exposure depth maps and our fused depth maps*

means of the well-known iterative closest point (ICP) algorithm [BM92, CM91]. We then compare the alignment error and the convergence.

In our experiment, we mount the camera on a professional tripod and capture a static indoor scene by rotating the tripod stepwise. We use six exposures for fusion, then rotate the tripod by 10 degrees and capture another series of exposures. For each position, we compute two point clouds. One directly from a single exposure with an optimal integration time, the second from the fused depth map. This results in two pairs of point clouds that have to be aligned. We use the ICP implementation from Kjer and Wilm [KW10] to determine a rigid transformation. For our fused solution the algorithm converges equally fast but the final error is smaller (see Figure 9).

Further we compared the resulting transformation with our manually defined ground truth - a rotation by 10 degrees around the y-axis. Here, the rotation error is defined as the deviation from the identity

$$e(R, R_1) = ||I - RR_1^T||_F,$$

where R is the assumed correct rotation and R_1 the one we test, while $||\bullet||_F$ denotes the Frobenius norm. Our fused solution results in an error of 0.0956 while the single exposure solution produces an error of 0.1598. This is an reduction of about 40%.

4.4. Limitations

Beside the shown positive examples our method is also limited. In some scenarios the fused depth maps are of equal quality as a single exposure. Table 1 shows that our method is not always far better than simpler approaches. This is the case if all objects have good (Lambertian) reflection properties and their distance is in a limited range. Our method works best if the distances and reflection properties are highly varying. However, neither strong noise in certain areas that are farer than the theoretic limit of 7.5 meters, nor severe over-saturation can be resolved properly. In addition, it is necessary that none of the input images is completely noisy.

5. Conclusion

We have presented and evaluated a new method to enhance the performance of time-of-flight imaging devices. We developed test methods that do not need any extra hardware like a laser scanner in order to estimate the quality of our method. Our method successfully fuses several exposures into a single depth map and is on one hand not limited in the number of exposures and on the other hand fast enough to perform in real-time.

Our method not only works for fusing depth maps captured with different integration times, it also allows the combination of images with other varying parameters like the modulation frequency. We expect our method to achieve even better results for future time-of-flight cameras with an extended theoretic range.

References

[BA83] BURT P., ADELSON E.: The laplacian pyramid as a compact image code. *IEEE Transactions on Communications 31*, 4 (Apr. 1983), 532–540. 3

[BM92] BESL P., MCKAY N.: A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence 14* (1992), 239–256. 6

[BOL*05] BÜTTGEN B., OGGIER T., LEHMANN M., KAUFMANN R., LUSTENBERGER F.: Ccd / cmos lock-in pixel for range imaging : Challenges , limitations and state-of-the-art, 2005. 2

[CM91] CHEN Y., MEDIONI G.: Object modeling by registration of multiple range images. In *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on* (apr 1991), pp. 2724 –2729 vol.3. 6

[CSC*10] CUI Y., SCHUON S., CHAN D., THRUN S., THEOBALT C.: 3d shape scanning with a time-of-flight camera. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (June 2010), IEEE, pp. 1173–1180. 2

[DM08] DEBEVEC P. E., MALIK J.: Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH 2008 classes on - SIGGRAPH '08* (New York, New York, USA, Aug. 2008), ACM Press, p. 1. 2

[FAACT10] FOIX S., ALENYÀ G., ANDRADE-CETTO J., TORRAS C.: Object modeling using a tof camera under an uncertainty reduction approach. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on* (may 2010), pp. 1306 –1312. 3

[FPH09] FRANK M., PLAUE M., HAMPRECHT F. A.: Denoising of continuous-wave time-of-flight depth images using confidence measures. *Optical Engineering 48*, 7 (2009), 077003. 3

[Gos05] GOSHTASBY A.: Fusion of multi-exposure images. *Image and Vision Computing 23*, 6 (June 2005), 611–618. 2, 4

[HA09] HAHNE U., ALEXA M.: Depth imaging by combining time-of-flight and on-demand stereo. In *Dynamic 3D Imaging* (2009), Kolb A., Koch R., (Eds.), vol. 5742 of *Lecture Notes in Computer Science*, Springer, pp. 70–83. 2

[HSJS10] HUHLE B., SCHAIRER T., JENKE P., STRASSER W.: Fusion of range and color images for denoising and resolution enhancement with a non-local filter. *Computer Vision and Image Understanding 114*, 12 (Dec. 2010), 1336–1345. 2

[KBKL09] KOLB A., BARTH E., KOCH R., LARSEN R.: Time-of-flight sensors in computer graphics. In *Eurographics State of the Art Reports* (2009), pp. 119–134. 1, 2

[KRI06] KAHLMANN T., REMONDINO F., INGENSAND H.: Calibration for increased accuracy of the range imaging camera swissranger. In *Proceedings of the ISPRS Commission V Symposium 'Image Engineering and Vision Metrology'* (Dresden, 2006), Maas H.-G., Schneider D., (Eds.), vol. XXXVI, pp. 136–141. 2

[KW10] KJER H., WILM J.: *Evaluation of surface registration algorithms for PET motion correction*. Master's thesis, Technical University of Denmark, 2010. 7

[LK07] LINDNER M., KOLB A.: Calibration of the intensity-related distance error of the pmd tof-camera. *Proceedings of SPIE* (2007), 67640W–67640W–8. 2

[LLK08] LINDNER M., LAMBERS M., KOLB A.: Sub-pixel data fusion and edge-enhanced distance refinement for 2d/3d images. *International Journal of Intelligent Systems Technologies and Applications 5*, 3/4 (2008), 344. 2

[LSKK10] LINDNER M., SCHILLER I., KOLB A., KOCH R.: Time-of-flight sensor calibration for accurate range sensing. *Computer Vision and Image Understanding 114*, 12 (Dec. 2010), 1318–1328. 2

[MB09] MALPICA W. S., BOVIK A. C.: Range image quality assessment by structural similarity. In *Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing* (Washington, DC, USA, 2009), ICASSP '09, IEEE Computer Society, pp. 1149–1152. 2, 4

[MDH*08] MAY S., DROESCHEL D., HOLZ D., WIESEN C., BIRLINGHOVEN S.: 3d pose estimation and mapping with time-of-flight cameras. *International Conference on Intelligent Robots and Systems (IROS), 3D Mapping workshop, Nice, France I*, September 2007 (2008), 2008–2008. 2

[MKV07] MERTENS T., KAUTZ J., VAN REETH F.: Exposure fusion. In *15th Pacific Conference on Computer Graphics and Applications (PG'07)* (Oct. 2007), IEEE, pp. 382–390. 2, 3, 4

[MWSP06] MAY S., WERNER B., SURMANN H., PERVOLZ K.: 3d time-of-flight cameras for mobile robotics. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Oct. 2006), Ieee, pp. 790–795. 1, 5

[OBL*05] OGGIER T., BÜTTGEN B., LUSTENBERGER F., BECKER G., RÜEGG B., HODAC A.: Swissranger sr3000 and first experiences based on miniaturized 3d-tof cameras, 2005. 1

[RDP*11] REYNOLDS M., DOBOŠ J., PEEL L., WEYRICH T., BROSTOW G. J.: Capturing time-of-flight data with confidence. In *CVPR* (2011). 3

[RWPD05] REINHARD E., WARD G., PATTANAIK S., DEBEVEC P.: *High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting (The Morgan Kaufmann Series in Computer Graphics)*. Morgan Kaufmann, 2005. 2

[STDT08] SCHUON S., THEOBALT C., DAVIS J., THRUN S.: High-quality scanning using time-of-flight depth superresolution. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on* (june 2008), pp. 1 –7. 2

[STDT09] SCHUON S., THEOBALT C., DAVIS J., THRUN S.: Lidarboost: Depth superresolution for tof 3d shape scanning. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* (June 2009), IEEE, pp. 343–350. 2

[VFM07] VINYALS O., FRIEDLAND G., MIRGHAFORI N.: Revisiting a basic function on current cpus: a fast logarithm implementation with adjustable accuracy, 2007. 4

[WBSS04] WANG Z., BOVIK A., SHEIKH H., SIMONCELLI E.: Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing 13*, 4 (Apr. 2004), 600–612. 4

[YIM07] YAHAV G., IDDAN G., MANDELBOUM D.: 3d imaging camera for gaming application. In *Consumer Electronics, 2007. ICCE 2007. Digest of Technical Papers. International Conference on* (jan. 2007), pp. 1 –2. 1

[ZWYD08] ZHU J., WANG L., YANG R., DAVIS J.: Fusion of time-of-flight depth and stereo for high accuracy depth maps. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008)* (2008). 2