https://doi.org/10.2352/ISSN.2470-1173.2017.20.3DIPM-003

A Magnifier on Accurate Depth Jumps

Thomas Hach, Arnold & Richter Cine Technik, Germany
Sascha Knob, Hochschule RheinMain, Germany

Abstract

RGBD cameras capturing color and depth information are highly promising for various industrial, consumer and creative applications. Among others, these applications are segmentation, gesture control or deep compositing. Depth maps captured with Time-of-Flight sensors, as a potential alternative to vision-based approaches, still suffer from low depth resolution. Various algorithms are available for RGB-guided depth upscaling but they also introduce filtering artifacts like depth bleeding or texture copying. We propose a novel superpixel-based upscaling algorithm, which employs an iterative superpixel clustering strategy to achieve improved boundary reproduction at depth discontinuities without aforementioned artifacts. Concluding, a rich ground-truth-based evaluation validates that our upscaling method is superior compared to competing state-of-the-art algorithms with respect to depth jump reproduction. Reference material is collected from a real RGBD camera as well as the Middlebury 2005 and 2014 data sets. The effectiveness of our method is also confirmed by usage in a depth-jump-critical computational imaging use case.

Introduction

Capturing RGB images and depth information has accessed numerous applications. Plenty of them like depth-image-based rendering [1], object tracking [2], image segmentation [3] or deep compositing [4] are improved or facilitated when a depth channel becomes available. In particular, upcoming computational imaging applications will highly benefit from the accompanied depth channel. Depth is typically acquired using passive triangulation techniques or active Time-of-Flight (TOF) imaging. While passive methods deliver higher-resolution depth maps with only minor noise, they lack in mechanical and computational simplicity as well as in depth density.

Contrary, Time-of-Flight sensors are cheap and easy to use in practical environments and deliver only low-resolution depths maps containing a noticeable amount of noise. Due to the practical benefits and the availability of a dense depth map, they are considered as promising. Hence, this paper concentrates on depth maps from TOF sensors. In order to overcome the remaining problem of the low resolution, RGB-guided depth map upscaling is typically applied. Therein, it is commonly assumed that depth discontinuities coincide with changes in the objects' color appearance. For example, the well-known joint bilateral filter (JBF) deploys color intensity distances to estimate depth jumps at their more precise locations in a matching color image [5].

One important aspect, which has not yet been handled appropriately in guided upscaling algorithms like the joint bilateral filter (JBF), noise-aware depth upscaling (NAFDU) filter and guided filter (GF) [5, 6, 7], is flying pixels. Flying pixels are explained by sampling artifacts of the low TOF sensor resolution. That is, due to the very low spatial sampling rate of TOF sensors, physical depth discontinuities generate disturbing mixed depth measurements from foreground and background at object boundaries, yielding a contradictory distance measurement of somewhere in between. Figure 1 underlines the urgency of this problem considering a JBF-filtered depth map projected as a point cloud in 3D space and the corresponding result of our approach. Applying guided upscaling to those raw depth maps will increase the amount of flying pixels at object boundaries since initial flying pixels are treated as valid inputs. Thus, in TOF imaging terminology, all pixels that are natively captured or processed to situate between two objects, while actually corresponding to either of the two surfaces are called flying pixels. At this point, we want to reveal the underlying problem. All guided upscaling methods have their origin in conventional color image filtering where the known rules of visual perception hold. Considering depth maps, visual assessment is not sensible although being often applied. For example, in depth maps, a pixel corresponding to an opaque object must have one unique depth value. Regarding the formation mechanism of flying pixels, this rule is violated. Guided upscaling filters, designed to be edge-aware, aim to reduce the number of those filtered pixels at boundaries. In visual imaging, this process leads to perceptually sharp edges. In depth sensing, this process also leads to perceptually sharp edges in the depth maps but these filters are not designed to assign distinct depth values to objects. Hence, a perceptually sharp edge in a depth map, when visualized and assessed like a grayscale image, still contains a tremendous amount of violating flying pixels. This destroys computational imaging applications like keying based on depth masks or lens synthesis among many others.

Thus, our contribution is a novel upscaling algorithm focusing on correct depth edge assignment and thus improved upscaling respecting the flying pixels. We utilize the high segmentation performance of superpixel (SP) algorithms to indicate depth edges which are assumed to coincide with the SP boundaries. This includes a novel seed growing strategy to overcome the underlying uncertainty relation between the uniqueness of the depth estimates within each SP cluster and granularity of recognized objects. All found depth discontinuities are finally fed into a modified joint bilateral filter which thereby allows for the generation of uniquely assigned pixels at depth edges.

Prior Art

Numerous color-guided depth upscaling approaches have been designed in the past.

The joint bilateral filter (JBF) comprises a modified version of the bilateral filter proposed by Kopf et al. [5, 8]. Herein, an output pixel is the weighted sum of its neighbors preserving edges by additionally considering RGB pixel intensity differences. Compared to the proposed algorithm, JBF is real-time compatible. The one-stage guided filter (GF1s) is designed to overcome JBF-specific artifacts like the gradient reversal effect [9, 10]. Except for edge reversal, the global behavior of the GF1s is similar to the one of JBF. The two-stage guided filter (GF2s) is an extension of the general guided filter with special focus on depth map enhancement [10]. Therein, Hui et al. proposed a two-stage strategy, which means enhancing depth using the RGB guide in the first stage and using the filtered depth output as guide in the second stage. This procedure retains all benefits of guided image filtering while re-

(a) JBF upscaling (b) our approach before JBF smoothing stage (c) our full approach

Figure 1: Comparison of resulting 3D point clouds, when using JBF upscaling and our approach with different configurations: the number of flying pixels is drastically reduced. The corresponding depth maps are given in Figure 9.

ducing, but not finally eliminating, texture copying and depth bleeding artifacts. The noise-aware depth upscaling filter (NAFDU) is based on the JBF [6]. Therein, Chan et al. provide an adaptive weighting between JBF and a conventional bilateral filter (BF) to reduce texture copying [8]. Yang et al. propose a cost volume approach inspired from stereo vision approaches yielding promising results especially in the case of reconstructing simple geometric primitives [11]. Our algorithm is also capable of providing the latter, however, we provide examples which are not easy to solve using a single color key like in their work.

Yang et al. incorporate bilateral filters in a median and bilateral filter fusion framework yielding the solution to an adaptive cost aggregation problem [12]. Therein, the bilateral filter preserves edges while the median makes the method more robust to noise.

Huhle et al. propose a non-local filter for upscaling and suppressing flying pixels [13]. Although the approach looks promising, they clearly lack experimental results and the approach is computationally highly demanding as 2x upscaling takes 14 minutes on a CPU. In comparison, our approach lasts between 4 and 5 minutes for 12x upscaling on a single CPU.

Constant time weighted median filtering proposed by Ma et al. [14] is based on weighted median filtering [15, 16]. This method is optimized for speed while preserving accuracy. All of the aforementioned filters will generate a noticeable amount of flying pixels during upscaling [17].

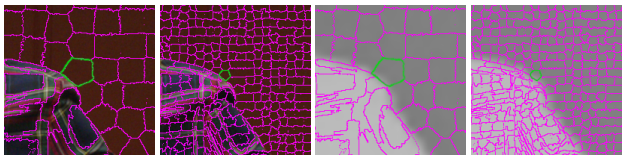
The following filters are based on iterative or optimization methods. They are computationally demanding, especially when using those methods for 16-bit depth maps. Based on our evaluation with full-HD 16-bit data, which is a very typical configuration in professional applications, the computation times strongly increase up to the range of hours. Unfortunately, this holds for all of the following algorithms because of the utilization of Markov-Random-Fields or TV solvers, which are optimizing discrete labels.

Yang et al. propose an iterative cost volume filtering for depth super resolution [11]. Therein, a 3D cost volume is updated based on the current depth map and consecutively filtered by the JBF. The total generalized variation (TGV) is based on standard total variation techniques. However, Ferstl et al. replaced the L1 norm, used as the standard in total variation approaches, by the L2 norm because the L1 norm enforces over-flattening of objects [18]. Additionally, they add an anisotropic diffusion tensor for enhanced control of the optimization direction. High-quality depth map upsampling based on non-local means filtering (NLM) is proposed by Park et al. [19]. Therein, the NLM term is extended by a typical smoothness and data term known from Markov-Random-Field (MRF) approaches. Pure MRFs are deployed by Diebel et al. [20] using specific weighting factors to a combined smoothness and data term. They, for instance state that they solve $2 \cdot 10^5$ nodes, which equals only 448x448 pixels, of 8-bit depth in 2 seconds. Our approach targets 1920x1080 of underlying 16-bit data, which increases the computation time tremendously. Joint geodesic upsampling, which is based on the all-pair-shortest-path problem, is introduced by Liu et al. [21]. Therein, they propose a novel approximation algorithm to mainly decrease the complexity. An adaptive auto-regressive model is employed by Yang et al. [22]. Their auto-regressive predictor operates pixel-wise and is designed in dependence on the local correlation of the raw depth map and the non-local similarity of the accompanied RGB image. Weighted mode filtering presented by Min et al. is based on a joint RGB and depth histogram [23]. While processing the histogram, each bin is weighted by the similarity between reference and neighboring pixels of the RGB image.

The following methods employ superpixel techniques, which is highly related to our approach. Soh et al. presented a method called superpixel-based depth image super-resolution (SDIS) which is the first incorporating the concept of superpixels in depth map upscaling [24]. They propose a superpixel segmentation on both, the RGB and naively upscaled depth map. Since the superpixel segmentation is performed in the color image and depth map, the difference in resolution leads to a consecutive sorting algorithm. All pixels within a certain distance W from the crucial depth map superpixel boundary are resorted in accordance to the superpixel membership of nearest neighbors in the RGB segmentation outside the blurry corridor W. After resorting, least-square planes are fitted in the depth of each superpixel yielding a piece-wise linear depth map. Eventually, this depth map is smoothed using a maximum a posteriori Markov-Random-Field (MAP-MRF) approach, which leads to a high computational burden and again introduces numerous flying pixels. Our own experiments show that they also suffer from a large scaling factor between RGB images and depth maps because the algorithm relies in parts on the superpixel segmentation from the upscaled depth image. Blurred depth edges will lead to an erroneous segmentation and a strong slanted edge phenomenon within the least-squares plane fitting. Thus, to our observation, SDIS works best if depth edges are already quite defined.

Matsuo and Aoki propose to merge the initial superpixel segmentation based on similarities in the principal component analysis of the latter patches [25]. Although their results are promising they clearly lack a method to process depth jumps. Their approach simply neglects steep slanted patches leading to invalid areas around objects as shown in the according paper. This underlines the effectiveness of our approach which is able to handle depth jumps appropriately.

Kim et al. propose an extended JBF filter which is guided by a so-called color segment set which corresponds to a superpixel segmentation of the RGB image [26]. Therein, they apply a user-defined threshold which limits the depth refinement to a certain



(a) (b) (c) (d)

Figure 2: High-resolution RGB images with high (a) and low (b) superpixel oversegmentation (superpixels are indicated by purple lines). According depth maps with the superpixel segmentation swapped from RGB (c,d). Small superpixels tend to contain only flying pixels at object boundaries while providing an improved boundary recall

depth range. Next to the hole filling task, we dismiss this approach due to the limitation of manually pre-selecting local boundaries by adjusting the depth range. In the remainder of the depth map, JBF is used which incorporates an additional depth range term.

Upscaling Algorithm

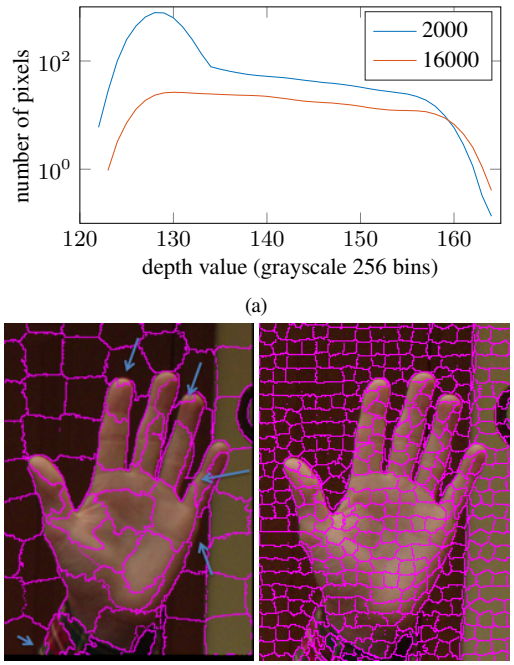
Our algorithm, called iterative superpixel-guided depth upscaling (ISUDU), is based on two observations. First, in the case of TOF cameras, confidence of a single pixel is said to be low because of high noise and measurement errors [27]. Thus, it is reasonable to increase a pixel's confidence by involving neighbors. We utilize superpixel clustering to find appropriate neighbors. Second, in conventional image segmentation as well as our previous paper [17], superpixel algorithms have proven to provide outstanding boundary recall, which is required in order to reach the design goal of distinct depth map edges [28, 29]. Still, there remains the uncertainty relation when applying those clusters to a depth map. An increasing number of superpixel clusters results in an increased boundary recall with the ability to resolve smaller objects while the number of depth pixels within each superpixel becomes smaller. Hence the confidence decreases. Figure 2 shows two superpixel segmentations at an object boundary using 2000 and 16000 superpixels per image. This translates to 1000 and 130 pixels per superpixel. The underlying depth map was upscaled and registered to the final resolution using bicubic interpolation. Figure 3a shows the comparison of smoothed depth histograms of the highlighted superpixels in Figure 2. The logarithmically scaled ordinate reveals that for the case of the larger superpixel (blue line), an obvious maximum can be found, whereas in the case of the smaller superpixel (red line) a clear maximum cannot be identified. The latter observation describes the case of a superpixel that mainly consists of flying pixels. Figure 3b and Figure 3c show the difference in boundary recall between a high and low number of superpixels. The blue arrows highlight boundary recall issues of larger superpixels.

Flying Pixel-Aware Depth Upscaling

An essential prerequisite for our approach is aligned RGB images and depth maps. Thus, our monocular RGBD camera approach is beneficial as the registration task can be solved robustly in 2D space [30, 31]. Favoring our camera is not a limitation of the algorithm but it is beneficial for the final results as we do not have to consider occlusion problems at the image areas of interest for upscaling, which are object boundaries.

Our algorithm starts with a high number of initial RGB superpixels, enforcing a high boundary recall.

To overcome the uncertainty relation when trying to transfer these clusters into the depth map, the core of our algorithm groups the



(a)

(b) (c)

Figure 3: (a) Larger superpixels at object boundaries provide a centroid of valid depth pixels (blue line) compared to small superpixels which are dominated by flying pixels (orange line); (b) RGB image with overlaid large superpixel segmentation; (c) small superpixel segmentation. Arrows indicate problems when using large superpixels considering boundary recall and small object representation;

small superpixels, resulting in larger superpixel clusters. This grouping is done deploying our superpixel growing method called iterative superpixel clustering (ISC). Therein, we employ the correlation between the color histograms of one superpixel and its neighbors. We assume that image regions of similar true depth share similar superpixel color histogram information. Using superpixel color histogram similarities, we are not restricted to pixel-wise color differences as used by the majority of RGB-guided filters but the evaluation of richer texture patch information. This procedure leads to larger superpixels with the same boundary recall and fine object segmentation as the initial segmentation while providing a higher number of pixels within each superpixel, especially at depth jumps.

In more detail, we use state-of-the-art SLIC superpixel segmentation [28]. SLIC yields superpixel clusters which provide strong compactness [17]. Hence, it is unlikely that superpixels have irregular shapes like slim but long shapes which is unwanted since non-compact long shapes could wriggle along object boundaries including again only flying pixels. Hence, using unconstrained superpixel methods is not compatible when using this ISC as there is a likely solution to merge multiple slim superpixels around a larger portion of an object outline. This effect contradicts with the idea to statistically minimize the influence of flying pixels.

ISC treats each superpixel sequentially. Therein, each superpixel's neighbors are determined and the histogram distance is computed using the correlations of the histograms. The histograms are built for each color channel using 1024 bins for 16-bit color data. The neighbor candidate with the highest cross correlation, which must be higher than a minimum c_{min}, is then examined in more detail. The

correlation coefficient between superpixel i and j of color channel $c \in \{R,G,B\}$ is calculated by

$$\rho(i,j,c) = \frac{1}{N-1} \sum_{n} \left(\frac{h^i_{n,c} - \mu_{h^i_c}}{\sigma_{h^i_c}} \right) \left(\frac{h^j_{n,c} - \mu_{h^j_c}}{\sigma_{h^j_c}} \right), \quad (1)$$

where $\mu_{h^i_c}$ and $\mu_{h^j_c}$ are the means and $\sigma_{h^i_c}$ and $\sigma_{h^j_c}$ are the standard deviations of the respective histograms h^i_c and h^j_c. The sum is iterated over the bins $n \in [0,N-1]$. The final histogram distance metric is then computed as

$$d(h^i,h^j) = \rho(i,j,R) \cdot \rho(i,j,G) \cdot \rho(i,j,B). \quad (2)$$

Moreover, we want to achieve strong compactness of potentially merged superpixel. Hence, we additionally constrain the resulting shape of a merged superpixel by limiting the number of pixels in the merged superpixel and by limiting the new radius. If a candidate fails, this superpixel neighbor is blacklisted for the current superpixel and hence ignored during upcoming iterations, when again searching for remaining neighbors. However, the failed superpixel stays available for different superpixels. ISC of the whole image is repeated iteratively while reducing the lower limit c_{min} for the cross correlation if any further neighbor candidates can be found due to all cross correlation coefficients lower than the current minimum. The step width of the correlation minimum reduction is 10% and the blacklist is cleared for all superpixels after each reduction of c_{min}. The reason for using the reduction principle is again enhanced robustness of the clustering process. If the correlation minimum c_{min} was to small in the beginning, there is merging of superpixels with a low correlation although a local superpixel with higher index might be a better candidate with higher cross correlation distance. This is due to the sequential processing of the superpixels in the order of their increasing index. Thus, starting with a high correlation minimum ensures that there is only merging of superpixels with a high correlation value first. If all high correlation merges are found, we reduce the minimum to allow for further merges down to a certain limit c_{end}. Using the reduction approach, the resulting global estimate of clustered superpixels are closer to optimality.

The signal flow of the ISC algorithm is also given in Figure 4. A limitation of ISC and the consecutive assignment of depth is obtaining larger clusters on smooth surfaces which can produce artificial depth jumps. Thus, we added a processing step called superpixel reconstruction, which compares the merged superpixels after ISC and the initial superpixel map. Therein, we again deploy properties of the depth map. The blue histogram in Figure 3a can be decomposed in two parts. First, there is a clear maximum which indicates a small extension in depth. Second, there is a shallow slope which represents the flying pixels. Thus, superpixels that only contain a distinct maximum are characterized by a small standard deviation. Hence, if a merged superpixel cluster contains an initial small superpixel whose depth standard deviation exceeds a threshold, the large superpixel is retained because it is likely that numerous flying pixels are contained. Otherwise, the initial small superpixel segmentation is used in the final superpixel map instead of the merged cluster. Thereby, we reduce artifacts in smooth areas like walls or tables. Figure 5 shows all three stages from the initial superpixel segmentation towards the final superpixel segmentation.

The final superpixel map is now transferred to the initially registered depth map Z^{BC}, where the depth-to-RGB image registration uses bicubic interpolation. For each superpixel area, the maximum likelihood (ML) depth is evaluated using histograms and assigned to the whole superpixel area in an intermediate depth map Z^{ML}, which is analogous to the step described in our previous work [17]. Similar to the findings of Soh et al., we still see small depth jumps between each superpixel region in the Z^{ML} depth map [24]. Contrary to Soh et al., we neither assign least-square (LS) fit planes within a superpixel nor use MAP-MRF estimators for the final depth. In contrast we assign constant depth values within a superpixel which avoids strongly slanted planes due to fitting LS planes at blurred depth boundaries.

Superpixel-Aware Joint Bilateral Filter

The computation of the final depth map starts using a modified JBF on Z^{BC}. The introduced modification consists of an adaptive awareness for larger depth jumps in Z^{ML}. Consequently, JBF provides beneficial smoothing in areas of Z^{ML} with small jumps while clear object boundaries with their depth jumps are transferred from Z^{ML}. Thereby, we benefit from the high segmentation performance of our previous flying pixel compensation approach (FPC) [17]. The modified JBF filter kernel F is now written as

$$F_q = \begin{cases} \mathrm{JBF}_q, & \text{if } I \bigcup II \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

with
case I: $\sigma\left(Z^{BC}_{SP(i)}\right) < \sigma_{max}$, and case II: $\left|Z^{ML}_q - Z^{BC}_p\right| < sp_{\text{tolerance}}$.

In Equation 3, coordinates $q \in \Omega$ describe the coordinates with respect to the kernel window Ω centered at pixel p. We designed the cases I and II, when conventional JBF is applied. Otherwise, we discontinuously force the kernel entry to 0, which leads to edges with only few flying pixels. The function $\sigma\left(Z^{BC}_{SP(i)}\right)$ evaluates the standard deviation of Z^{BC} within area of superpixel i to design the filter noise-aware. $\left|Z^{ML}_q - Z^{BC}_p\right|$ prevents the emergence of depth steps at superpixel border areas within an actually smooth surface.

Utilizing this modified JBF filter kernel F results in smoothed surfaces and an improved refinement of edges while retaining the beneficial properties of the JBF elsewhere. In this configuration, the edge enforcement by the Gaussian distance cost function for intensities of the standard JBF settings can be relaxed because the superpixel part mostly takes care of it. This is highly beneficial since texture copying in smooth areas can thus be avoided.

Finally, to compute the output depth Z', the depth input is locally adapted at regions which are most-likely to contain depth discontinuities by switching the depth input value Z_q in

$$Z'_p = \frac{1}{k} \sum_{q \in \Omega} Z_q \cdot F_q \quad (4)$$

feeding Z^{BC}, if the distance between Z^{ML} and Z^{BC} at pixel q is smaller than the noise level expressed as standard deviation $\sigma\left(Z^{BC}_{SP(i)}\right)$ in $SP(i)$ and feeding Z^{ML} otherwise by

$$Z_q = \begin{cases} Z^{BC}_q, & \text{if } \left|Z_{SP,q} - Z_q\right| < \sigma\left(Z^{BC}_{SP(i)}\right) \\ Z^{ML}_q, & \text{otherwise.} \end{cases} \quad (5)$$

Thereby, we arrive at a strong depth jump segmentation. Moreover, these modifications of the standard JBF ensure that first, Z^{BC} pixels situated in adjacent superpixel regions which differ too much from the

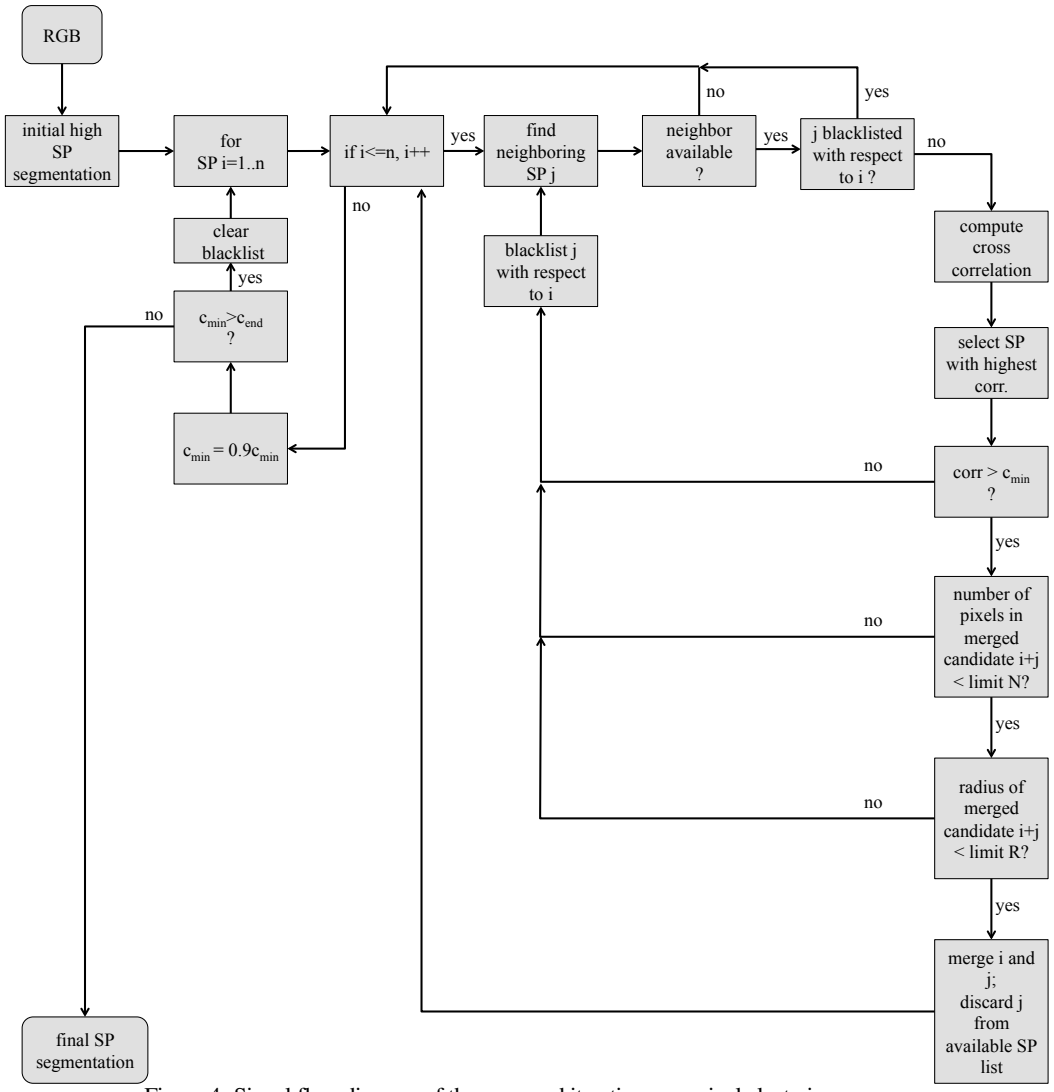


Figure 4: Signal flow diagram of the proposed iterative superpixel clustering process

(a) (b) (c)

Figure 5: (a) Initial RGB segmentation with small superpixels; (b) Result after ISC; (c) Reconstruction of small superpixels on smooth depth regions

Z^{ML} estimate are dismissed by Equation 3. This reduces the number of flying pixels because of *avoiding weak filter weights*. Second, Equation 5 ensures that Z^{ML} depth values are used instead of Z^{BC} depth values at object boundaries since these are distinct compared to the blurry ones in the Z^{BC} depth map. Thereby we *avoid incorporating flying pixels* in the weighted sum yielding the resulting depth map.

Evaluation

In this work, we provide an evaluation on real RGBD data as well as on the widely used Middlebury data set [32, 33]. Real TOF depth data is necessary because we observe principle-based differences in depth data characteristics compared to Middlebury. In detail, we encounter noise types like range ambiguity or Poisson noise as well as measurement errors [27, 34, 32, 35, 36].

Ground Truth and Metrics

For the evaluation of the pure depth upscaling performance, we adopt our novel metric called noise-aware depth edge classification (NADEC) [17].

Therein, a single RGB frame from an RGBD camera can be used as input for quick manual rotoscoping. Optional partitioning of the rotoscoped objects provides a series of ground truth patches, which can be obtained from one single image. Furthermore, one opaque pixel must correspond to one depth value. Semi-transparent RGB pixels cannot match sensible depth values. Hence, for ground truth selection, all image parts with in-focus depth jumps are suitable. Figure 8 depicts such rotoscoped binary masks. Therein, object boundaries derived from the RGB image are correct up to minimal blur which arouses from remaining low pass properties of physical lenses, even when they are fully focused, and low pass properties of filter glasses in front of the image sensor. Hence, semi-transparency is also induced by mixture of foreground and background colors, similar to the occurrence of native flying pixels. We account for this minimal blur at boundaries, which does not correspond to semi-transparencies, by drawing the mask boundary in the visual center of the blurred RGB region. However, these pixels also mark the accuracy limit of the metric.

Next, we define a criterion that reveals the depth segmentation performance of depth boundaries. Therefore, we use the bicubicly upscaled raw depth map and the mask from rotoscoping the depth patch to segment into foreground and background. The mean value and the standard deviation are calculated for each of the two regions separately. As a numerical measure, our metric utilizes typical classification metrics. In this case, it is the number of false positives f_p and the number of false negatives f_n.

Considering depth maps, false negative pixels are depth values of the background located in areas of the foreground whereas false positive pixels consist of depth values of the foreground object bleeding into the background area. Subsequent classification is done next. Depth values outside a range of $\mu \pm c \cdot \sigma$ of the respective area are declared as false.

NADEC is mathematically given by

$$NADEC = \frac{f_n + f_p}{N}. \tag{6}$$

where N normalizes to the total number of pixels in the image. Hence, we obtain 0, as best possible value, if and only if all depth pixels share perfect segmentation and thus best upscaling with respect to the ground truth mask. Figure 6 visualizes the idea behind NADEC. NADEC assumes a typical depth jump with flying pixels in between.

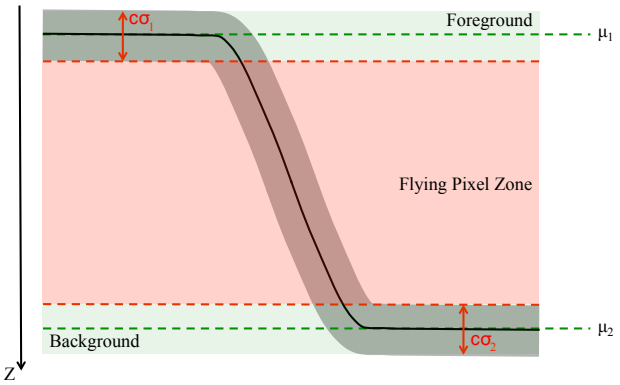


Figure 6: Illustration of our proposed NADEC metric for upscaling performance evaluation. The figure shows a typical native edge jump, especially when registered to the high-resolution RGB image grid, with flying pixels between foreground and background as indicated by the green and red zones. NADEC determines the estimated means for foreground and background as well as its standard deviations. Thereby a corridor for valid pixels can be built.

For each region, the foreground and background, the means μ_1, μ_2 and the standard deviations σ_1, σ_2 are estimated to yield a corridor for valid depth values while respecting noise. Contrary, NADEC identifies a zone where flying pixels are most likely and thus false positives and false negatives in these regions can be counted. These add up to the flying pixels found in the opposing green area.

Synthetic Ground Truth Data

Next to the evaluation using real camera data, we compare our algorithm employing several images of the well-known Middlebury data sets. Therefore, we downscale the ground truth depth map by a factor of 12, which corresponds to typical scaling requirements when using a state-of-the-art PMD TOF sensor with 160x90 pixels and full-HD RGB with 1920x1080 pixels. Then, we apply an AWGN noise model with a standard deviation of 0.032 for Middlebury 2005 and 0.0094 for Middlebury 2014 to simulate TOF noise. These values are based on real noise measurements [36][27]. The value of 0.032 corresponds to 24 cm whereas the value of 0.0094 corresponds to 7 cm, which are typical noise observations.

For numerical evaluation, we use NADEC and RMSE as metrics. RMSE is possible because a ground truth depth map is available in the synthetic case. However, we still prefer NADEC over RMSE as it provides more information about the upscaling performance for the discussed reasons.

Results

Utilizing the proposed metrics, we evaluated our approach against numerous state-of-the-art algorithms. For nearest-neighbor (NN), bilinear (BL), bicubic (BC) interpolation as well as GF1s and GF2s the standard MATLAB implementations are used. JBF [6], NAFDU [6], weighted mode filtering (WMF) [23] and spatial depth image superresolution (SDIS) [24] have been re-implemented in Matlab using mex extensions. In the case of SDIS, the iterative conditional modes (ICM) algorithm is used for solving the MAP-MRF [37]. The parameters of each algorithm had been optimized towards minimum NADEC error in the case of the *body* segment, before evaluating the sub-patches. For TGV, we use the MATLAB code

and parameters provided by Ferstl et al. [18]. Table 2 summarizes the results using the real RGBD data sample. Our method outperforms all competing algorithms with respect to depth jump reconstruction.

Computational demands are also evaluated by classification of the computation time of all methods in 3 categories. These are denoted with the tags (A), (B) and (C), referring to processing times less than 5 minutes (A), less than an hour (B) and more than one hour (C) per depth map, when processing 1920x1080 pixels of 16-bit data in non-GPU C/C++ on a Core i7 2.2GHz processor. Additionally, computation time strongly depends on the captured scene in terms of texture and depth structure. Figure 9 depicts all depth maps generated by the evaluated algorithms in their optimized state.

For the 2014 Middlebury data samples, we provide RMSE and NADEC in Tables 3 and 4 to compare the average effectiveness on the whole image (RMSE) and the pure depth jump segmentation on a subset of selected regions (NADEC), which are marked by red rectangles in Figure 11. The results are also depicted in Figure 11. Our algorithm does not lead to strong outliers as shown by the RMSE evaluation, where our approach performs equivalent to state-of-the-art NAFDU. However, our algorithm noticeably outperforms NAFDU regarding the depth jump reconstruction shown by the NADEC error. In particular, the NAFDU algorithm has been run through a time-consuming parameter optimization whereas the parameters of our algorithm have been found by quick heuristics. The list of parameters used for ISUDU is given in Table 1.

Computational Depth-of-Field Synthesis

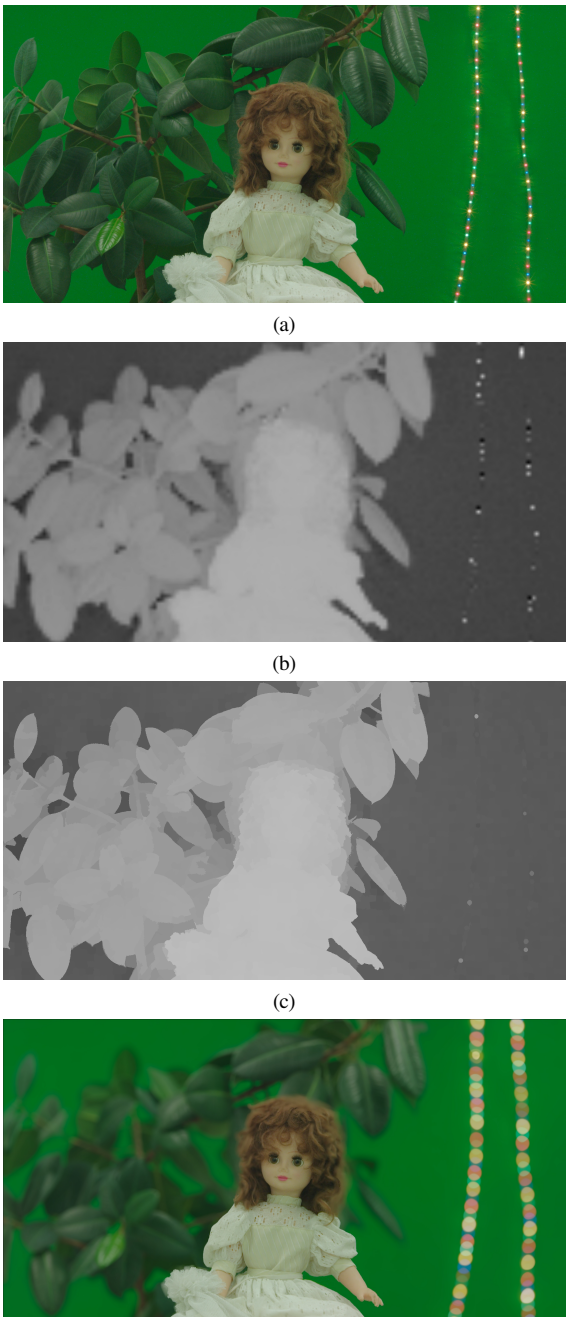
Next to numerical evaluation, we demonstrate our algorithm within a computational imaging application. We chose our recently published algorithm [38], which allows for cine or DSLR-style depth-of-field synthesis of large lens apertures .

The synthesis method requires very precise depth boundaries for the in-focus objects to convincingly separate defocused and focused areas. In particular, the human eye is very sensitive to minor issues at focused boundaries. A noticeable amount of flying pixels, which is produced by other methods, will lead to soft edges of the in-focus objects in the rendered RGB image. Figure 7c depicts the depth map processed by our ISUDU. Figure 7d demonstrates that the rendered result, when using our upscaling approach, provides accurately segmented RGB object boundaries derived from the upscaled depth map without noticeable edge blur. Soft edges at semi-transparent or very tenuous objects, especially the hair of the doll, are due to wrong TOF depth measurements, which is a limitation of low-resolution native TOF sensing rather than a limitation of the depth-of-field renderer or ISUDU algorithm.

Conclusion

In this work, we proposed a novel depth map superresolution algorithm based on an RGB superpixel oversegmentation strategy. We apply our iterative clustering procedure to yield an outstanding edge segmentation at depth discontinuities. Our method is texture adaptive, which results in differently sized superpixel clusters that are deployed in an improved joint bilateral kernel. This leads to upscaled depth maps with superior edge segmentation compared to state-of-the-art algorithms. This property is especially beneficial when looking for a solution to master a typical depth map artifact called flying pixels.

The twofold evaluation conducted in this paper numerically proves the effectiveness of our algorithm on both, real RGBD and a selection of the Middlebury data sets. Therein, special focus lied on the evaluation of the depth jump segmentation.



(a) (b) (c) (d)

Figure 7: Cine-style depth-of-field and Bokeh synthesis when using our upscaling method for the required depth map: (a) All-in-focus image; (b) Raw depth map; (c) Upscaled depth (ISUDU); (d) Cine-style depth-of-field synthesis using our algorithm [38] based on (c)

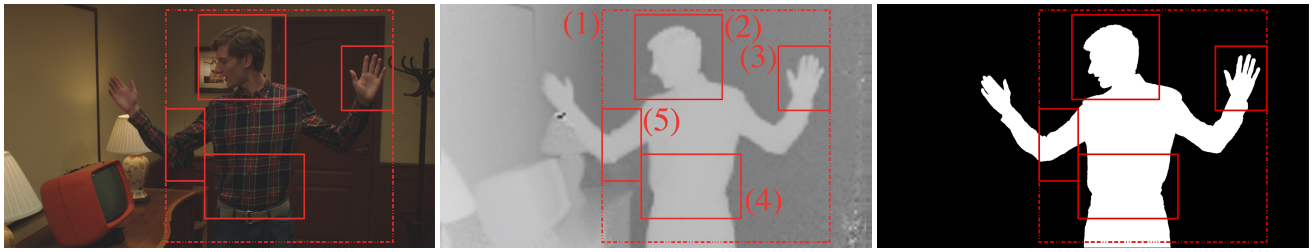
Utilizing depth maps processed by our method in a recent high-quality DSLR lens synthesis algorithm, we yield visually compelling results for the first time due to its superior boundary separation.

References

[1] Yang, X., Liu, J., Sun, J., Li, X., Liu, W., Gao, Y.: DIBR-Based View Synthesis For Free-Viewpoint Television. In: 3DTV

Table 1: List of parameters used for ISUDU

Parameter	Value
number of bin for histograms	1024
number of superpixels	7000
SLIC compactness	10
c_{min}	0.9
c_{end}	0.65
maximum radius	300
maximum pixels	99
JBF σ_{max}	0.01
JBF $sp_{tolerance}$	0.05
JBF window	21
JBF σ_S	10
JBF σ_R	0.05



(a) (b) (c)

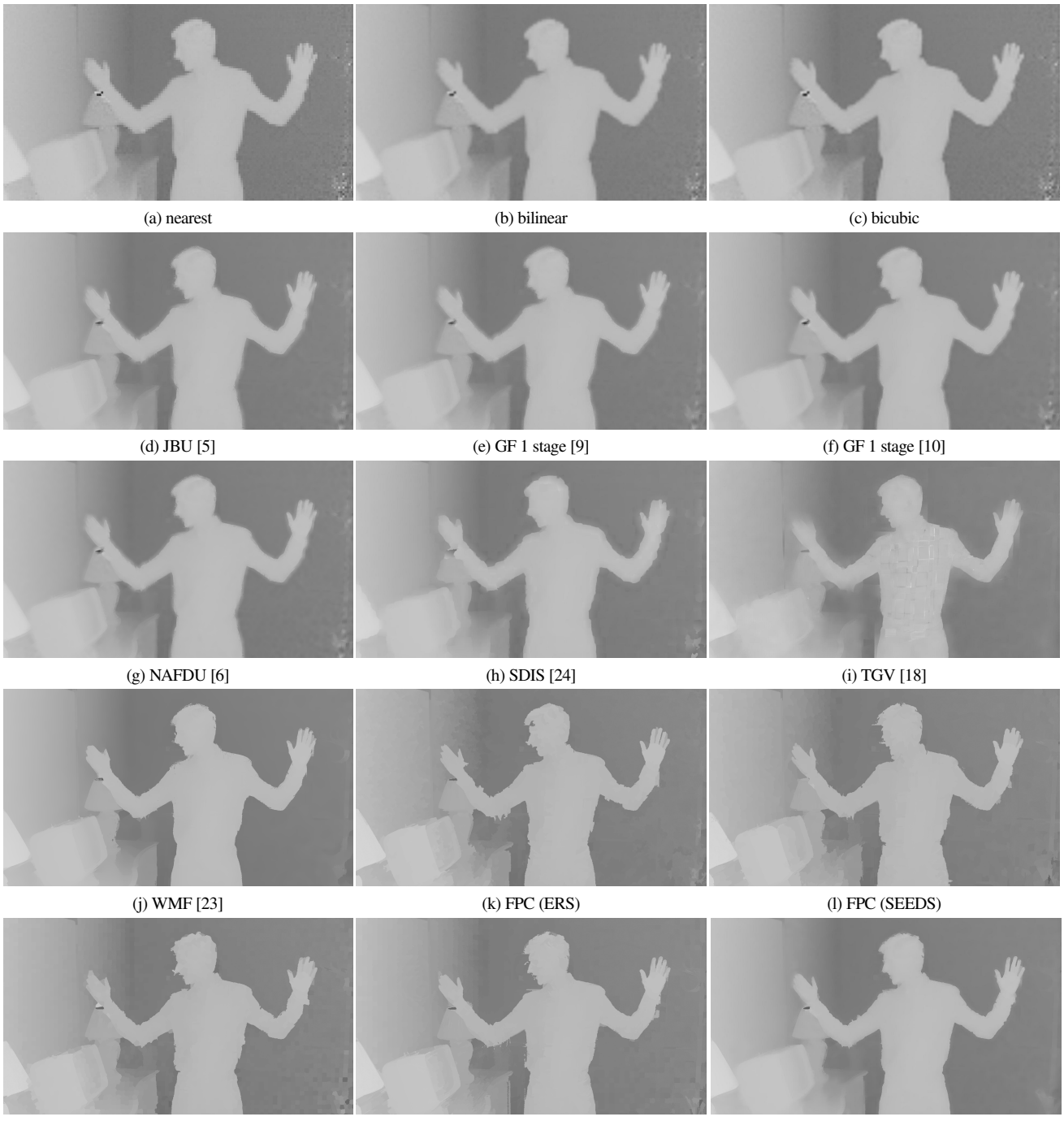
Figure 8: RGB image of the acting scene (a) [17] with bicubicly upscaled depth map (b) and segmentation mask (c) for the human actor. Test patches: (1, dashed line) body, (2) head, (3) hand, (4) waist, (5) elbow

Table 2: State-of-the-art algorithms in comparison to our approach using the acting scene as introduced in Figure 8. Benchmark was generated employing the proposed metric NADEC ($c=1$) [%] (best values of each column are marked in bold face; second best value in italic face)

Method	body	head	hand	elbow	waist	avg.
nearest (A)	1.57	0.30	0.17	0.11	0.17	0.46
bilinear (A)	1.73	0.32	0.14	0.12	0.18	0.50
bicubic (A)	1.59	0.30	0.13	0.10	0.16	0.47
JBF [5] (A)	1.31	0.29	0.12	0.064	0.17	0.39
NAFDU [6] (B)	1.34	0.28	0.12	0.067	0.16	0.39
GF_1s [9] (A)	1.37	0.29	0.10	0.082	0.16	0.40
GF_2s [10] (A)	1.38	0.28	0.11	0.075	0.13	0.40
SDIS [24] (C)	1.13	0.24	0.17	0.085	0.26	0.38
TGV [18] (B)	1.36	0.26	0.13	0.11	0.26	0.42
$WMF_{(8\text{-bit})}$ [23] (C)	0.60	0.26	0.13	**0.028**	**0.061**	*0.23*
FPC with CRS (ours) (A)	*0.59*	*0.19*	0.10	0.044	*0.084*	**0.20**
FPC with ERS (ours) (A)	0.71	*0.19*	*0.087*	0.042	0.13	*0.23*
FPC with SEEDS (ours) (A)	0.72	**0.18**	0.13	0.038	0.077	*0.23*
FPC with SLIC (ours) (A)	0.70	0.21	0.12	*0.032*	0.095	*0.23*
ISUDU (ours) (A)	**0.58**	**0.18**	**0.084**	0.035	0.13	**0.20**

Table 3: RMSE of Middlebury 2014 images: scaling 12, $\sigma=0.032$ (best values per column in bold face)

Method	Adirondack	Jadeplant	Motorcycle	Piano	Pipes	Playroom	Playtable	Recycle	Shelves	Vintage	Average
nearest	0.00988	0.0132	0.0241	0.00779	0.0681	0.0135	0.0114	0.00848	0.00920	0.00718	0.0173
bilinear	0.00826	0.0115	0.0218	0.00633	0.0622	0.0117	0.00969	0.00712	0.00782	0.00599	0.0152
bicubic	0.00820	0.0114	0.0215	0.00656	0.0623	0.0117	0.00971	0.00740	0.00798	0.00629	0.0153
NAFDU [6]	0.00699	**0.0107**	**0.0219**	0.00531	0.0665	**0.0107**	**0.00801**	0.00573	0.00671	0.00573	**0.0148**
ISUDU (ours)	**0.00603**	0.0123	0.0238	**0.00506**	**0.0646**	0.0118	0.00919	**0.00532**	**0.00648**	**0.00510**	0.0149



(a) nearest (b) bilinear (c) bicubic

(d) JBU [5] (e) GF 1 stage [9] (f) GF 1 stage [10]

(g) NAFDU [6] (h) SDIS [24] (i) TGV [18]

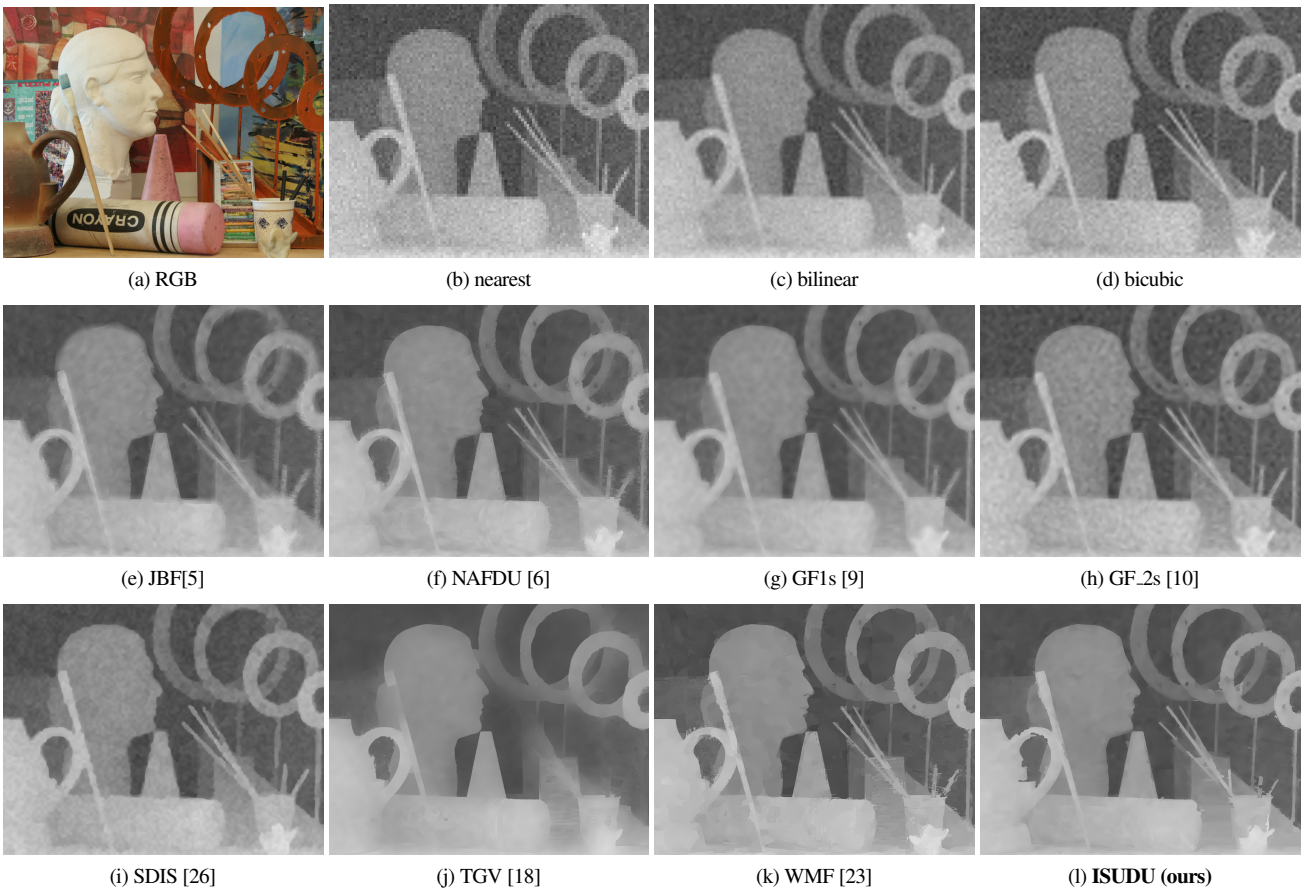
(j) WMF [23] (k) FPC (ERS) (l) FPC (SEEDS)

(m) FPC (SLIC) (n) FPC (CRS) (o) **ISUDU (ours)**

Figure 9: Resulting depth maps of state-of-the-art filters, our flying pixel removal approach FPC with all 4 different superpixel algorithms and the result of our proposed ISUDU

Table 4: NADEC error ($c=1$) in permille of Middlebury 2014 images: scaling 12, $\sigma=0.032$

Method	Adir1	Adir2	Adir3	Adir4	Playr1	Playr2	Playt1	Playt2	Playt3	Avg.
nearest	8.50	4.53	14.03	3.01	2.98	2.83	1.31	2.50	2.41	4.68
bilinear	5.91	3.77	10.76	2.74	1.81	2.19	1.07	2.06	1.95	3.58
bicubic	7.17	4.18	12.25	3.01	2.20	2.50	1.12	2.31	2.13	4.10
NAFDU [6] (RMSE)	1.97	**3.70**	7.95	0.55	0.85	**2.07**	**0.67**	0.54	**1.00**	2.14
ISUDU (ours)	**1.16**	3.87	**5.42**	**0.45**	**0.40**	2.11	0.83	**0.36**	1.33	**1.77**



(a) RGB (b) nearest (c) bilinear (d) bicubic

(e) JBF[5] (f) NAFDU [6] (g) GF1s [9] (h) GF_2s [10]

(i) SDIS [26] (j) TGV [18] (k) WMF [23] (l) **ISUDU (ours)**

Figure 10: Upscaling results using the Art image of the Middlebury 2005 data set [32]; Simulated scaling factor of 12 corresponding to our system and added AWGN noise with $\sigma = 0.032 \widehat{=} 24$cm.

Conference: The True Vision-Capture, Transmission and Display of 3D Video. (2011) 1–4

[2] Bevilacqua, A., Di Stefano, L., Azzari, P.: People Tracking Using a Time-of-Flight Depth Sensor. In: IEEE Conf. Video and Signal Based Surveillance. (2006) 89–89

[3] Wang, L., Zhang, C., Yang, R., Zhang, C.: Tofcut: Towards robust real-time foreground extraction using a time-of-flight camera. In: Proc. 5th Int. Symposium 3D Data Processing, Visualization and Transmission. (2010)

[4] Lokovic, T., Veach, E.: Deep Shadow Maps. In: Proc. 27th Annu. Conference on Computer Graphics and Interactive Techniques, ACM Press/Addison-Wesley Publishing Co. (2000) 385–392

[5] Kopf, J., Cohen, M.F., Lischinski, D., Uyttendaele, M.: Joint bilateral upsampling. ACM Transactions on Graphics **26**(3) (2007) 96

[6] Chan, D., Buisman, H., Theobalt, C., Thrun, S.: A noise-aware filter for real-time depth upsampling. In: Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications-M2SFA2. (2008)

[7] He, K., Sun, J., Tang, X.: Guided image filtering. In: European Conference on Computer Vision. (2010) 1–14

[8] Tomasi, C., Manduchi, R.: Bilateral Filtering For Gray and Color Images. In: 6th Int. Conf. Computer Vision. (1998) 839–846

[9] He, K., Sun, J., Tang, X.: Guided Image Filtering. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(6) (2013) 1397–1409

[10] Hui, T.W., Ngan, K.N.: Depth enhancement using RGB-D guided filtering. In: IEEE International Conference on Image Processing. (2014) 3832–3836

[11] Yang, Q., Yang, R., Davis, J., Nister, D.: Spatial-Depth Super Resolution for Range Images. In: IEEE Conference on Computer Vision and Pattern Recognition. (2007) 1–8

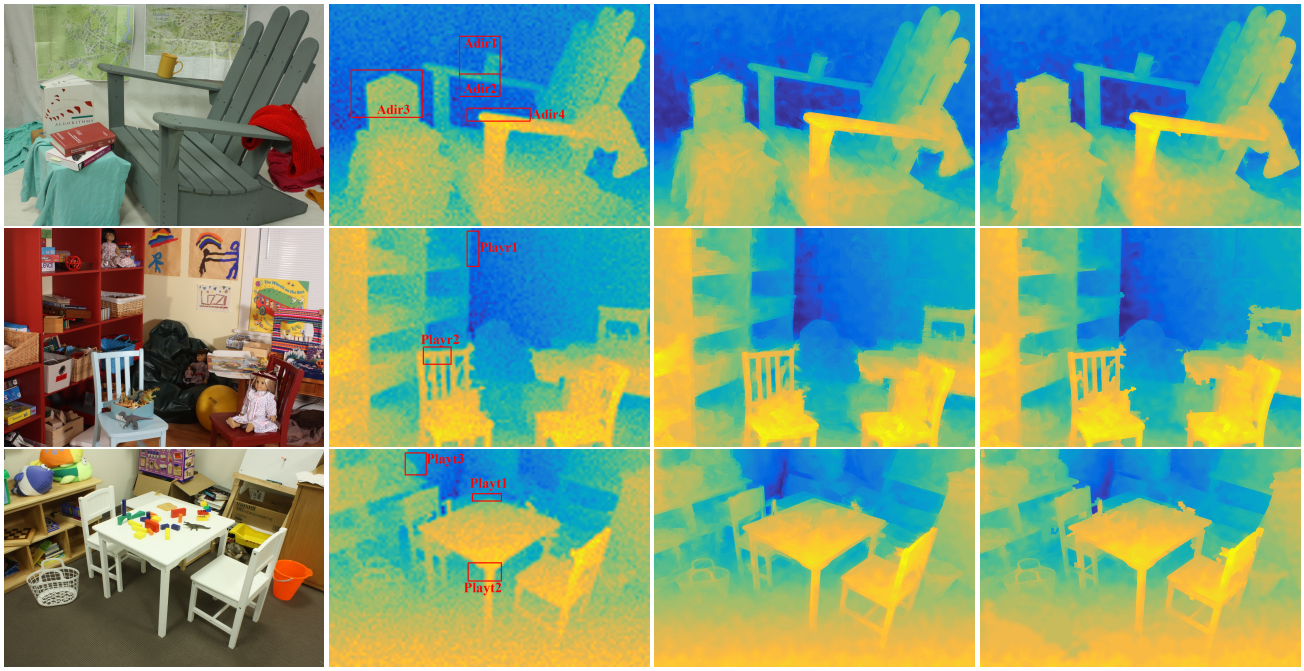
[12] Yang, Q., Ahuja, N., Yang, R., Tan, K.H., Davis, J., Culbertson, B., Apostolopoulos, J., Wang, G.: Fusion of Median and Bilateral Filtering for Range Image Upsampling. IEEE Trans. Image Process. **22**(12) (2013) 4841–4852

[13] 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) (2010) 1336–1345

[14] Ma, Z., He, K., Wei, Y., Sun, J., Wu, E.: Constant Time Weighted Median Filtering for Stereo Matching and Beyond. In: 2013 IEEE International Conference on Computer Vision. (2015) 49–56

[15] Rhemann, C., Hosni, A., Bleyer, M., Rother, C., Gelautz, M.: Fast cost-volume filtering for visual correspondence and beyond. In: IEEE Conf. Computer Vision and Pattern Recognition. (2011) 3017–3024

[16] Sun, D., Roth, S., Black, M.J.: Secrets of optical flow estimation and their principles. In: IEEE Conf. Computer Vision and



RGB | input depth | optimized NAFDU [6] | **ISUDU (ours)**

Figure 11: Evaluation on 3 images of the Middlebury 2014 data set [33]: (first column) low resolution depth maps (scaling 12, $\sigma = 9.43 \cdot 10^{-3} \widehat{=} 7$cm); (second column) upscaled depth maps using an RMSE-optimized NAFDU [6] and (last column) upscaled depth maps after our method: Notice, all depth maps are in false color representation which gives a clearer view on the amount of flying pixels revealed as light blur at object boundaries when comparing NAFDU and our approach.

Pattern Recognition. (2010) 2432–2439

[17] Hach, T., Knob, S., Steurer, J.: High-Fidelity Time-of-Flight Edge Sampling Using Superpixels. In: Electronic Imaging. (2016) 1–9

[18] Ferstl, D., Reinbacher, C., Ranftl, R., Ruether, M., Bischof, H.: Image Guided Depth Upsampling using Anisotropic Total Generalized Variation. In: Proc. Int. Conf. on Computer Vision. (2013)

[19] Park, J., Kim, H., Tai, Y.W., Brown, M.S., Kweon, I.: High quality depth map upsampling for 3D-TOF cameras. In: IEEE International Conference on Computer Vision. (2011) 1623–1630

[20] Diebel, J., Thrun, S.: An Application of Markov Random Fields to Range Sensing. In: Neural Information Processing Systems Conference. Volume 5. (2005) 291–298

[21] Liu, M.Y., Tuzel, O., Taguchi, Y.: Joint Geodesic Upsampling of Depth Images. In: IEEE Conf. Computer Vision and Pattern Recognition. (2013) 169–176

[22] Jingyu, Y., Xinchen, Y., Kun, L., Chunping, H., Yao, W.: Color-Guided Depth Recovery From RGB-D Data Using an Adaptive Autoregressive Model. IEEE Trans. Image Process. **23**(8) (2014) 3443–3458

[23] Min, D., Lu, J., Do, M.N.: Depth Video Enhancement Based on Weighted Mode Filtering. IEEE Transactions on Image Processing **21**(3) (2012) 1176–1190

[24] Soh, Y., Sim, J.Y., Kim, C.S., Lee, S.U.: Superpixel-based depth image super-resolution. In: IS&T/SPIE Electronic Imaging. (2012) 82900D–10

[25] Matsuo, K., Aoki, Y.: Depth Interpolation via Smooth Surface Segmentation Using Tangent Planes Based on the Superpixels of a Color Image. In: 2013 IEEE International Conference on Computer Vision Workshops. (2014) 29–36

[26] Kim, S.Y., Cho, J.H., Koschan, A., Abidi, M.: Spatial and Temporal Enhancement of Depth Images Captured by a Time-of-Flight Depth Sensor. 20th International Conference on Pattern Recognition (2010) 2358–2361

[27] Edeler, T. In: Bildverbesserung von Time-Of-Flight Tiefenkarten. Shaker, Aachen (2012) 12–15

[28] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Susstrunk, S.: SLIC superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Analysis and Machine Intelligence **34**(11) (2012) 2274–2282

[29] Reso, M., Jachalsky, J., Rosenhahn, B., Ostermann, J.: Temporally Consistent Superpixels. In: IEEE Int. Conf. Computer Vision. (2013) 385–392

[30] Hach, T., Steurer, J.: A Novel RGB-Z camera for High-Quality Motion Picture Applications. In: 10th European Conference on Visual Media Production. (2013) 1–10

[31] Hach, T., Seybold, T.: Spatio-Temporal Denoising for Depth Map Sequences. International Journal of Multimedia Data Engineering and Management **7**(2) (2016)

[32] Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: IEEE Conf. Computer Vision and Pattern Recognition. (2007) 1–8

[33] Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X., Westling, P.: High-resolution stereo datasets with subpixel-accurate ground truth. In: Pattern Recognition. (2014) 31–42

[34] Hirschmuller, H., Scharstein, D.: Evaluation of Cost Functions for Stereo Matching. In: IEEE Conference on Computer Vision and Pattern Recognition. (2007) 1–8

[35] Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G.,

Nesic, N., Wang, X., Westling, P. In: High-Resolution Stereo Datasets with Subpixel-Accurate Ground Truth. (2014) 31–42
[36] Hach, T., Seybold, T., Böttcher, H.: Phase-Aware Candidate Selection for Time-of-Flight Depth Map Denoising. In: IS&T/SPIE Electronic Imaging. (2015) 93930E–93930E–9
[37] Besag, J.: On the Statistical Analysis of Dirty Pictures. Journal of the Royal Statistical Society. Series B (Methodological) **48**(9) (1986) 259–302
[38] Hach, T., Steurer, J., Amruth, A., Pappenheim, A.: Cinematic Bokeh Rendering for Real Scenes. In: Proceedings of the 12th European Conference on Visual Media Production. (2015) 1–10

Author Biography

Thomas Hach received his B.Sc. and his Dipl.-Ing. degree from the Technical University of Munich (TUM) in 2011 and 2013, respectively. He is working as part of the R&D group of Arnold & Richter Cine Technik in Germany, where he currently finishes his doctorate. His ongoing work includes research on computational imaging algorithms, panoramic imaging, three-dimensional capture technologies, multi-camera as well as multi-modal sensor fusion approaches. He is author of several journal and conference papers in the field of camera systems, image processing, 3D sensing and visual effects.

Sascha Knob received his B.E. in media engineering from the University of Applied Science Wiesbaden Rüsselsheim, Germany (2015). He developed his bachelor thesis in the research and development division at ARRI in Munich and worked in the field of 3D imaging. His thesis includes the application of superpixel algorithms to depth maps. Presently, he is a master student at the University of Applied Science Wiesbaden Rüsselsheim.