

# Defocus Deblurring and Superresolution for Time-of-Flight Depth Cameras

Lei Xiao<sup>2,1</sup> Felix Heide<sup>2,1</sup> Matthew O’Toole<sup>3</sup> Andreas Kolb<sup>4</sup> Matthias B. Hullin<sup>5</sup>  
Kyros Kutulakos<sup>3</sup> Wolfgang Heidrich<sup>1,2</sup>

<sup>1</sup>KAUST <sup>2</sup>University of British Columbia <sup>3</sup>University of Toronto <sup>4</sup>University of Siegen  
<sup>5</sup>University of Bonn

## Abstract

Continuous-wave time-of-flight (ToF) cameras show great promise as low-cost depth image sensors in mobile applications. However, they also suffer from several challenges, including limited illumination intensity, which mandates the use of large numerical aperture lenses, and thus results in a shallow depth of field, making it difficult to capture scenes with large variations in depth. Another shortcoming is the limited spatial resolution of currently available ToF sensors.

In this paper we analyze the image formation model for blurred ToF images. By directly working with raw sensor measurements but regularizing the recovered depth and amplitude images, we are able to simultaneously deblur and super-resolve the output of ToF cameras. Our method outperforms existing methods on both synthetic and real datasets. In the future our algorithm should extend easily to cameras that do not follow the cosine model of continuous-wave sensors, as well as to multi-frequency or multi-phase imaging employed in more recent ToF cameras.

## 1. Introduction

Fast and high-quality depth-sensing cameras are highly desirable in mobile robotics, human-machine interfaces, quality control and inspection, and advanced automotive applications. Among the wide variety of depth-sensing technologies available (depth from stereo, structured lighting, Lidar scanning, *etc.*), continuous-wave time-of-flight (ToF) cameras have emerged as an efficient, low-cost, compact, and versatile depth imaging solution.

The active light source required to produce these ToF images presents significant drawbacks, however. To create a ToF image with high signal-to-noise ratio (SNR), the light signal must be sufficiently intense to overcome sensor noise and quantization effects. Factors determining the signal strength include light source power, integration time, imaging range, and lens aperture. In practice, light power is often limited for eye safety and energy considerations, and the

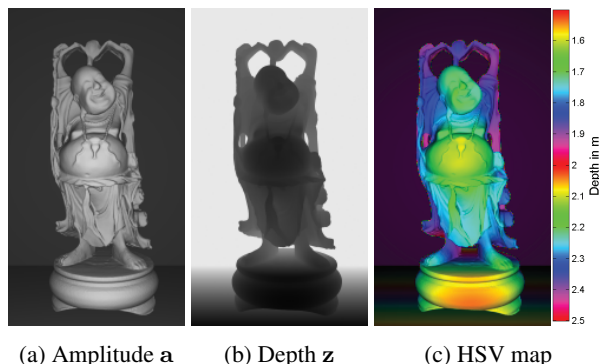


Figure 1: Simulated Buddha scene. The HSV map uses  $a$  as value and  $z$  as hue. The value and hue visualizes the magnitude and scaled phase of the complex-valued  $a \circ g(z)$  in Eq. (1), respectively.

integration time must be short enough to allow real-time operation. Consequently, ToF systems typically use imaging optics with large numerical aperture to make the best use of available light. However, this also comes at a cost; large apertures have a shallow depth of field and hence introduce defocus blur in the raw ToF images. Due to the non-linear image formation model of these cameras (see below), the depth of field blur presents a significant problem for ToF-cameras, generating artifacts such as “flying pixels” around depth discontinuities as well as loss of texture detail.

In this paper, we address this problem by introducing a new computational method to simultaneously remove defocus blur and increase the resolution of off-the-shelf ToF cameras. We do this by solving a semi-blind deconvolution problem, where prior knowledge of the blur kernel is available. Unlike past ToF deblurring techniques, our approach applies sparse regularizers directly to the latent amplitude and depth images, and supports deblurring ToF images captured with multiple frequencies, phases or exposures.

Continuous-wave ToF sensors are designed to have an image formation model that is linear in amplitude  $a$ , but non-linear in depth  $z$ , such that the captured raw sensor data

is given as

$$\mathbf{a} \circ \mathbf{g}(\mathbf{z}) \approx \mathbf{a} \circ e^{i\left(\frac{4\pi f}{c} \cdot \mathbf{z}\right)}, \quad (1)$$

where  $\circ$  represents component-wise multiplication of two vectors,  $f$  represents the frequency of the continuous-wave modulation, and  $c$  is the constant speed of light. The function  $\mathbf{g}(\mathbf{z})$  can either be calibrated [12], or, more commonly, is simply approximated by the complex-valued function from Eq. 1 (“cosine model”) [10]. Fig. 1 shows a simulated scene with visualization.

We aim to compute a solution to the following ill-posed inverse problem introduced in [9]:

$$\mathbf{b} = \mathbf{S}\mathbf{K}(\mathbf{z})(\mathbf{a} \circ \mathbf{g}(\mathbf{z})), \quad (2)$$

where the complex-valued vector  $\mathbf{b}$  represents the raw ToF measurements, the real-valued matrix  $\mathbf{S}$  is a downsampling operator, and the real-valued matrix  $\mathbf{K}(\mathbf{z})$  represents the spatially-varying blur kernel for a given depth map  $\mathbf{z}$ . The problem is ill-posed because the matrix  $\mathbf{S}\mathbf{K}(\mathbf{z})$  is usually not invertible, and semi-blind because the matrix  $\mathbf{K}(\mathbf{z})$  is known at each depth  $\mathbf{z}$ . In past work,  $\mathbf{S}$  is assumed to be the identity matrix.

It is important to note that Eq. (2) becomes the conventional image deblurring problem when  $f = 0$ , such that  $\mathbf{a} \circ \mathbf{g}(\mathbf{z}) = \mathbf{a}$ . Estimating the amount of defocus blur from a blurry amplitude map  $\mathbf{a}$  is a particularly challenging problem in this case, requiring either multiple photos or specialized optics [19]. Unlike conventional cameras, ToF cameras provide additional depth information that can be used to recover the defocus blur kernel much more robustly [9].

Our paper focuses on solving Eq. (2) to recover the deblurred amplitude and depth maps from a single blurry image, captured with an off-the-shelf ToF camera. Because this inverse problem is still an ill-conditioned problem, it’s critical to choose appropriate regularizers to reflect prior information on the solution (*i.e.*, sparse edges). Godbaz *et al.* [9] proposed differential priors that operate on the complex ToF image representing the cosine model, but it remains unclear what a good regularizer should even look like in this space. We instead choose to regularize our solution in the amplitude and depth map space directly. This introduces certain numerical challenges, because of the highly non-linear relation between the depth components and the raw ToF measurements (Eq. (1)). We relax this problem by splitting the optimization procedure into two parts, alternating between optimizing for amplitude and depth. Our method can seamlessly include a super-resolution component, helping to overcome the limited sensor resolution in current generation ToF cameras. Unlike earlier approaches, our method is also not inherently limited to the cosine model, and could be easily extended to calibrated waveforms in the future.

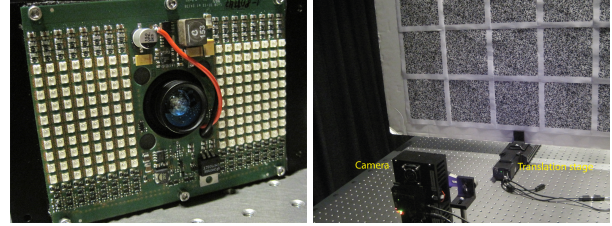


Figure 2: (a) PMD-Digicam camera. (b) Scene setup for PSF calibration.

## 2. Related work

Most existing ToF enhancement methods take as input the naive depth map from the camera software, rather than the raw complex-valued measurements. LidarBoost [22] and KinectFusion [14] utilized captures from multiple view-points to increase the depth resolution. Zhu *et al.* [24] explored the complementary characteristics of ToF and stereo geometry methods and combined them to produce better-quality depths. Other methods ([21, 15, 6]) used a high-resolution RGB or intensity image to guide the upsampling of the low-resolution depth map, based on the assumption about the co-occurrence of image discontinuities in RGB and depth data. These methods assume the scenes are all in-focus, while in this paper we deal with the scenes degraded by defocus blur. Another, these methods require additional hardware or multi-view captures, while as a complementary our method uses single-view captures from a single ToF sensor.

Godbaz *et al.* [8] proposed a two-stage method for parametric blind deconvolution of full-field continuous-wave Lidar imaging. They estimate the lens parameters from a pair of Lidar measurements taken at different aperture settings, and then deconvolve these complex-domain measurements (*i.e.*,  $\mathbf{a} \circ \mathbf{g}(\mathbf{z})$  in Eq. (2)), from which the final depth map is computed. Godbaz *et al.* [9] applied the coded aperture technique to extend the depth of field for full-field continuous-wave Lidar imaging. The complex-domain Lidar measurement is iteratively deconvolved with a simple Gaussian derivative prior, while at each iteration the blur kernel of each pixel is updated according to the currently estimated Lidar image. These two methods are close to ours in the sense of directly working on the raw measurements. In contrast to these methods which aim to deblur the complex measurements, our approach directly estimates the latent amplitude and depth from the degraded measurements. This allows us to apply separate regularizations on the amplitude and depth, and also supports for the next generation ToF cameras with multiple modulation frequencies, phases and exposures [10, 16].

Single-frequency ToF cameras have limited unambiguous distance range. Objects separated by the integer multi-

ples of the full range are indistinguishable. The next generation of ToF cameras use multiple modulation frequencies and phases to reduce the ambiguity [10, 16]. The multi-frequency/phase data can also help resolve “flying pixels” (mixtures of foreground and background depth) at the object boundaries, and suppress artifacts due to global illumination [7]. The ToF data captured with single exposure could be noisy or saturated due to scene properties such as surface materials and reflectivity. Multiple exposures are proposed to increase the dynamic range of the measurements and remove those unreliable pixels [10, 11]. Our algorithm adapts well to these cameras, since it directly estimates the latent amplitude and depth from raw measurements that could come from multiple sequential captures.

### 3. Proposed algorithm

#### 3.1. Algorithm overview

Given the raw measurements  $\mathbf{b}$  from a single view, our algorithm aims to remove optical lens blur and produce high quality depth map  $\mathbf{z}$  and amplitude map  $\mathbf{a}$ . The latent depth  $\mathbf{z}$  and amplitude  $\mathbf{a}$  are coupled in the measurements, thus we solve them as a joint optimization problem.

$$(\mathbf{a}, \mathbf{z}) = \underset{\mathbf{a}, \mathbf{z}}{\operatorname{argmin}} \|\mathbf{b} - \mathbf{SK}(\mathbf{a} \circ \mathbf{g}(\mathbf{z}))\|_2^2 + \Phi(\mathbf{a}) + \Psi(\mathbf{z}) \quad (3)$$

Eq. (3) shows the objective function we aim to minimize. The quadratic term represents a data-fitting error, assuming zero-mean Gaussian noise in the measurements.  $\Phi(\mathbf{a})$  and  $\Psi(\mathbf{z})$  represent regularizers for amplitude  $\mathbf{a}$  and depth  $\mathbf{z}$  respectively. The algorithm alternatively estimates  $\mathbf{a}$  and  $\mathbf{z}$ , and update the blur kernel matrix  $\mathbf{K}$  at the end of each iteration according to currently estimated  $\mathbf{z}$ .

Sparse gradient priors [19] have been widely used in natural image deblurring, but are improper for depth where the gradient could be non-zero for most pixels. The second-order derivative or sparse Laplacian priors have been used in surface denoising [3, 2, 23], but fail to model the discontinuity at object boundaries and thus does not distinguish the blurred and latent sharp depth. In this paper, we use the second-order total generalized variation (TGV<sup>2</sup>, [5]) for both the amplitude and depth, as shown in Eq. (4) and (5).

$$\Phi(\mathbf{a}) = \min_{\mathbf{y}} \lambda_1 \|\nabla \mathbf{a} - \mathbf{y}\|_1 + \lambda_2 \|\nabla \mathbf{y}\|_1 \quad (4)$$

$$\Psi(\mathbf{z}) = \min_{\mathbf{x}} \tau_1 \|\nabla \mathbf{z} - \mathbf{x}\|_1 + \tau_2 \|\nabla \mathbf{x}\|_1 \quad (5)$$

The TGV<sup>2</sup> prior automatically balances the first and second order derivative constraints. Following Knoll *et al.* [17], this can be intuitively understood as follows. In flat regions of  $\mathbf{z}$ , the second order derivative  $\nabla^2 \mathbf{z}$  is locally

---

#### Algorithm 1 Defocus Deblurring for ToF Depth Camera

---

**Input:** Raw measurements:  $\mathbf{b}$ ; modulation frequencies:  $\mathbf{f}$ ; upsampling ratio:  $r$ ; number of iterations:  $N$

**Output:** Estimated depth  $\mathbf{z}$  and amplitude  $\mathbf{a}$

- 1:  $\mathbf{a} = \operatorname{upsampling}(\operatorname{magnitude}(\mathbf{b}), r)$
- 2:  $\mathbf{z} = \operatorname{upsampling}(\operatorname{phase}(\mathbf{b})/(4\pi\mathbf{f}/c), r)$
- 3: **for**  $n = 1$  to  $N$  **do**
- 4:    $\mathbf{K} = \operatorname{updateKernel}(\mathbf{z})$
- 5:    $\mathbf{c} = \underset{\mathbf{c}}{\operatorname{argmin}} \|\mathbf{b} - \mathbf{SK}\mathbf{c}\|_2^2 + \rho \|\mathbf{c} - \mathbf{a} \circ \mathbf{g}(\mathbf{z})\|_2^2$
- 6:    $\mathbf{a} = \underset{\mathbf{a}}{\operatorname{argmin}} \rho \|\mathbf{c} - \mathbf{a} \circ \mathbf{g}(\mathbf{z})\|_2^2 + \Phi(\mathbf{a})$
- 7:    $\mathbf{z} = \underset{\mathbf{z}}{\operatorname{argmin}} \rho \|\mathbf{c} - \mathbf{a} \circ \mathbf{g}(\mathbf{z})\|_2^2 + \Psi(\mathbf{z})$
- 8: **end for**

---

small, thus it benefits the minimization problem in Eq. (5) to choose  $\mathbf{x} = \nabla \mathbf{z}$ , and minimize the second order derivative  $\|\nabla \mathbf{x}\|_1$ . While in the sharp edges of  $\mathbf{z}$  (*i.e.*, at object boundaries),  $\nabla^2 \mathbf{z}$  is larger than  $\nabla \mathbf{z}$ , thus it benefits to choose  $\mathbf{x}$  as zero, and minimize the first order derivative  $\|\nabla \mathbf{z}\|_1$ . Similar analysis applies for  $\mathbf{a}$  as well. The parameters  $\lambda_1, \lambda_2, \tau_1, \tau_2$  define the relative weights of the first and second order constraints. A modified version of TGV<sup>2</sup> was used in Ferstl *et al.* [6] for image guided depth upsampling.

$\mathbf{a} \circ \mathbf{g}(\mathbf{z})$  in Eq. (1) is highly nonlinear regarding to  $\mathbf{z}$ . To reduce the computation complexity in this nonlinear problem, the algorithm splits the data-fitting term in the objective (Eq. (3)) into a linear least square and a pixel-wise separable nonlinear least square (LSQ), as in Eq. (6). The scalar  $\rho$  defines the relative weight of the splitting term.

$$(\mathbf{a}, \mathbf{z}) = \underset{\mathbf{a}, \mathbf{z}, \mathbf{c}}{\operatorname{argmin}} \underbrace{\|\mathbf{b} - \mathbf{SK}\mathbf{c}\|_2^2}_{\text{linear LSQ for } \mathbf{c}} + \underbrace{\rho \|\mathbf{c} - \mathbf{a} \circ \mathbf{g}(\mathbf{z})\|_2^2}_{\text{separable nonlinear LSQ for } \mathbf{z}} + \Phi(\mathbf{a}) + \Psi(\mathbf{z}) \quad (6)$$

Algo. 1 shows the high-level pseudocode of the proposed method. The amplitude  $\mathbf{a}$  and depth  $\mathbf{z}$  are initialized as the magnitude and phase of the complex-valued measurement  $\mathbf{b}$ , respectively, and upsampled by nearest-neighbor method if superresolution wanted. Then the algorithm iteratively updates the blur kernel matrix  $\mathbf{K}$ , slack variable  $\mathbf{c}$ , amplitude  $\mathbf{a}$  and depth  $\mathbf{z}$ . The number of iterations  $N$  is typically set as 10. Details of each subproblem are described in Section 3.2-3.5.

#### 3.2. Kernel Update

The blur kernel is pre-calibrated at each pixel and sampled depth. The algorithm updates  $\mathbf{K}$  by a simple interpolated lookup in the pre-calibrated table of kernels according to the currently estimated depth  $\mathbf{z}$ . The details of the calibration procedure are explained in Section 4.

### 3.3. Slack Variable Update

The update of the slack variable  $\mathbf{c}$  requires solving a linear least square problem (Algo. 1, Line 5). Since the resulting linear equation system is positive-definite, a number of options for efficient solvers exist.

### 3.4. Amplitude Update

---

#### Algorithm 2 Update amplitude

---

**Input:**  $\mathbf{c}$ ,  $\mathbf{A}$ ,  $\rho$ ,  $\rho_a$ ,  $\lambda_1$ ,  $\lambda_2$ , number of iterations:  $M$

**Output:** Estimated amplitude  $\mathbf{a}$

- 1: **for**  $n = 1$  to  $M$  **do**
  - 2:  $\mathbf{a} = \underset{\mathbf{a}}{\operatorname{argmin}} \rho \|\mathbf{c} - \mathbf{A}\mathbf{a}\|_2^2 + \lambda_1 \rho_a \|\nabla \mathbf{a} - \mathbf{y} - \mathbf{p}_1 + \mathbf{u}_1\|_2^2$
  - 3:  $\mathbf{y} = \underset{\mathbf{y}}{\operatorname{argmin}} \lambda_1 \|\nabla \mathbf{a} - \mathbf{y} - \mathbf{p}_1 + \mathbf{u}_1\|_2^2 + \lambda_2 \|\nabla \mathbf{y} - \mathbf{p}_2 + \mathbf{u}_2\|_2^2$
  - 4:  $\mathbf{p}_1 = \underset{\mathbf{p}_1}{\operatorname{argmin}} \|\mathbf{p}_1\|_1 + \rho_a \|\nabla \mathbf{a} - \mathbf{y} - \mathbf{p}_1 + \mathbf{u}_1\|_2^2$
  - 5:  $\mathbf{p}_2 = \underset{\mathbf{p}_2}{\operatorname{argmin}} \|\mathbf{p}_2\|_1 + \rho_a \|\nabla \mathbf{y} - \mathbf{p}_2 + \mathbf{u}_2\|_2^2$
  - 6:  $\mathbf{u}_1 = \mathbf{u}_1 + \nabla \mathbf{a} - \mathbf{y} - \mathbf{p}_1$
  - 7:  $\mathbf{u}_2 = \mathbf{u}_2 + \nabla \mathbf{y} - \mathbf{p}_2$
  - 8: **end for**
- 

By substituting  $\Phi(\mathbf{a})$  into the update rule for the amplitude (Algo. 1, Line 6), we obtain the following optimization problem

$$\min_{\mathbf{a}, \mathbf{y}} \rho \|\mathbf{c} - \mathbf{A}\mathbf{a}\|_2^2 + \lambda_1 \|\nabla \mathbf{a} - \mathbf{y}\|_1 + \lambda_2 \|\nabla \mathbf{y}\|_1, \quad (7)$$

where  $\mathbf{A}$  is a diagonal matrix composed of  $\mathbf{g}(\mathbf{z})$ . This problem is solved by the alternating direction method of multipliers (ADMM [4]), as shown in Algo. 2. The  $\mathbf{a}$  and  $\mathbf{y}$  updates are linear least squares problems. The  $\mathbf{p}_{1,2}$ -updates are soft shrinkage problems and have closed form solutions [4]. The number of ADMM iterations  $M$  is typically set to 20. More details of each subproblem are provided in the supplementary document.

### 3.5. Depth Update

In a similar fashion, we can substitute  $\Psi(\mathbf{z})$  into the update rule for the depth (Algo. 1, Line 7), and obtain the optimization problem

$$\min_{\mathbf{z}, \mathbf{x}} \rho \|\mathbf{c} - \mathbf{a} \circ \mathbf{g}(\mathbf{z})\|_2^2 + \tau_1 \|\nabla \mathbf{z} - \mathbf{x}\|_1 + \tau_2 \|\nabla \mathbf{x}\|_1 \quad (8)$$

Once again, we apply the ADMM method to reduce this problem into easier subproblems, as shown in Algo. 3. For the sparse nonlinear least squares problem of updating  $\mathbf{z}$  (Algo. 3, Line 2), we use the Levenberg-Marquardt algorithm [18, 20] with an analytical Jacobian for the cosine model. To adapt our method to arbitrary (calibrated) waveforms, the only required change would be to replace this

---

#### Algorithm 3 Update depth

---

**Input:**  $\mathbf{c}$ ,  $\mathbf{a}$ ,  $\rho$ ,  $\rho_x$ ,  $\tau_1$ ,  $\tau_2$ , number of iterations:  $M$

**Output:** Updated depth  $\mathbf{z}$

- 1: **for**  $n = 1$  to  $M$  **do**
  - 2:  $\mathbf{z} = \underset{\mathbf{z}}{\operatorname{argmin}} \rho \|\mathbf{c} - \mathbf{a} \circ \mathbf{g}(\mathbf{z})\|_2^2 + \tau_1 \rho_x \|\nabla \mathbf{z} - \mathbf{x} - \mathbf{q}_1 + \mathbf{v}_1\|_2^2$
  - 3:  $\mathbf{x} = \underset{\mathbf{x}}{\operatorname{argmin}} \tau_1 \|\nabla \mathbf{z} - \mathbf{x} - \mathbf{q}_1 + \mathbf{v}_1\|_2^2 + \tau_2 \|\nabla \mathbf{x} - \mathbf{q}_2 + \mathbf{v}_2\|_2^2$
  - 4:  $\mathbf{q}_1 = \underset{\mathbf{q}_1}{\operatorname{argmin}} \|\mathbf{q}_1\|_1 + \rho_x \|\nabla \mathbf{z} - \mathbf{x} - \mathbf{q}_1 + \mathbf{v}_1\|_2^2$
  - 5:  $\mathbf{q}_2 = \underset{\mathbf{q}_2}{\operatorname{argmin}} \|\mathbf{q}_2\|_1 + \rho_x \|\nabla \mathbf{x} - \mathbf{q}_2 + \mathbf{v}_2\|_2^2$
  - 6:  $\mathbf{v}_1 = \mathbf{v}_1 + \nabla \mathbf{z} - \mathbf{x} - \mathbf{q}_1$
  - 7:  $\mathbf{v}_2 = \mathbf{v}_2 + \nabla \mathbf{x} - \mathbf{q}_2$
  - 8: **end for**
- 

derivative estimate with a tabulated version based on the calibration data. We use the cosine model for the experiments in Section 5 for fair comparisons with previous work, which makes the same assumption. Again, the  $\mathbf{q}_{1,2}$  updates are soft shrinkage problems, we use  $M = 20$  iterations, and all further details are provided in the supplementary document.

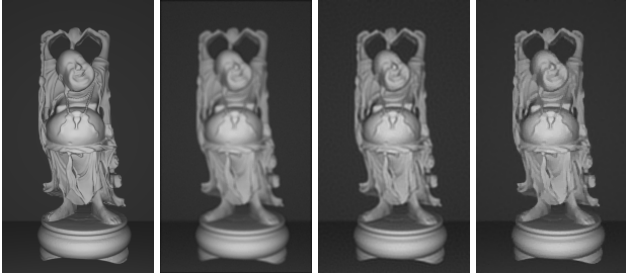
## 4. Calibration

The blur kernel (PSF) pre-calibration for real measurements is done by a similar approach to Heide *et al.* [13]. Fig. 2 shows the experiment setup. Printed random noise patterns are attached on a flat white board, which is held on a translation stage. The translation stage moves the white board to place from 60cm to 160cm away from the camera, with 1cm incremental. At each place, the camera captures with large aperture (the same aperture used for real measurements). Then, this process is repeated but with a small aperture so that the scene is nearly in-focus. Next, the amplitude images of the two captures at each place are used to estimate the PSF as a non-blind deconvolution problem.

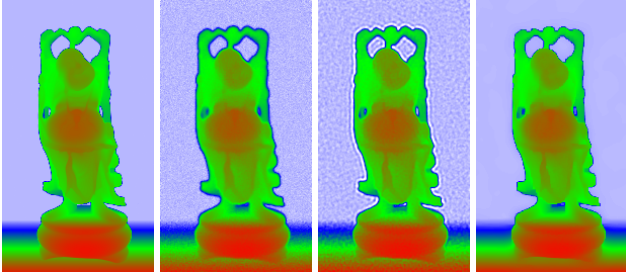
## 5. Results

We test the proposed algorithm on both synthetic and real datasets, and compare with two methods: the naive method, which computes the amplitude and depth as the magnitude and phase of the raw complex images respectively; and Godbaz *et al.* [9], which alternatively updates blur kernels and deconvolves the complex image with a Gaussian prior.

**Synthetic data.** The results on a simulated Buddha scene is shown in Fig. 3. The naive amplitude  $\mathbf{a}$  and depth  $\mathbf{z}$  are blurry and contain strong noise and flying pixels. In the visualized depth map, the flying pixels appear in dif-



(a) Estimated  $\mathbf{a}$ , from left to right, by ground truth, naive method (29.1dB), Godbaz method (31.9dB) and ours (34.5dB).



(b) Estimated  $\mathbf{z}$ , from left to right, by ground truth, naive method (36.2dB), Godbaz method (37.3dB) and ours (43.0dB).

Figure 3: Results on simulated Buddha scene with 0.5% white noise. Our method significantly reduce the blurriness and suppress noise in both  $\mathbf{a}$  and  $\mathbf{z}$ , and reducing the flying pixels at object boundaries. PSNR of the results are provided in the brackets.

ferent color than the foreground and background surface at the boundaries. Godbaz *et al.* method is highly sensitive to noise. Their result to some extent reduces the blurriness, but contains obvious noise, ringing artifacts and flying pixels. The proposed method significantly reduces the blurriness and flying pixels, and suppress the noise in both the amplitude  $\mathbf{a}$  and depth  $\mathbf{z}$ . The results are compared with ground truth data. Our approach produces much higher PSNR than the other methods.

In Fig. 4, we show the PSNR values of our estimated  $\mathbf{a}$  and  $\mathbf{z}$  at each iteration (i.e.,  $n$  in Algo. 1).

**Real data.** We captured real datasets using the Digicam camera from PMDTechnologies (Fig. 2) with a 6-15mm and  $f/1.4$  lens. 30MHz modulation frequency and 300 microsecond exposure time are used, and a single frame is captured for each scene. We crop out the pixels near image boundaries and the typical resolution of input images is  $250 \times 180$  pixels. The pre-calibrated PSFs have a width of 5-11 pixels at depths between 0.6 and 1.6m.

In Fig. 5 we show the photographs of the captured scenes. The results and comparisons are shown in Fig. 6, 9 and 12, and cropped regions are shown in Fig. 7, 10 and 13. In Fig. 8, we show the mesh geometry color-coded according to surface normal to better illustrate the depth results.

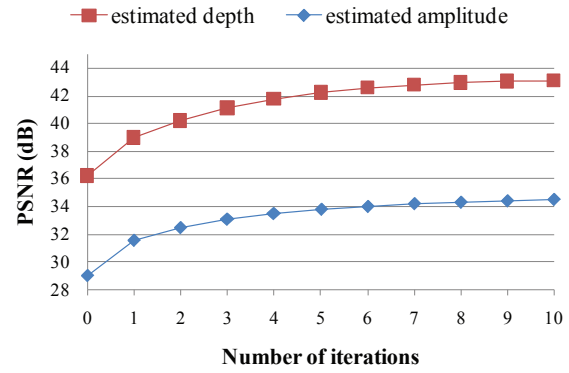


Figure 4: PSNR of our estimated amplitude  $\mathbf{a}$  and depth  $\mathbf{z}$  on the Buddha scene at each iteration.



(a) Camel

(b) Character



(c) Board

Figure 5: RGB Photos of the real scenes.

Please zoom in for better views.

Similarly as in the synthetic example, Godbaz *et al.* method is unable to handle noise (which is common in low-end ToF cameras), and fails to recover sharp scene features. Our method produces much higher quality amplitude and depth, in term of suppressing the noise, recovering sharp features and reducing flying pixels. We also run our algorithm with 2x superresolution (i.e., upsampling ratio  $r = 2$  in Algo. 1), and show that we achieve even better results with superresolution in our joint optimization framework. We use bicubic interpolation for the downsampling operator  $\mathbf{S}$ . In Fig. 11, we show our results with different upsampling ratios, and observe that little more details are recovered beyond 4x upsampling for our dataset.

We tune the parameters to generate results of compared method and ours. For our results,  $\rho$  (in Algo. 1),  $\rho_a$  (in Algo. 2) and  $\rho_x$  (in Algo. 3) are fixed as 0.125, 10 and 10 respectively. In Algo. 2,  $\lambda_1$  is typically set as 0.001 or 0.002, and  $\lambda_2$  as 20 or 40 times  $\lambda_1$ . In Algo. 3,  $\tau_1$  is typically set as 0.0005 or 0.001, and  $\tau_2$  as 20 or 40 times  $\tau_1$ .

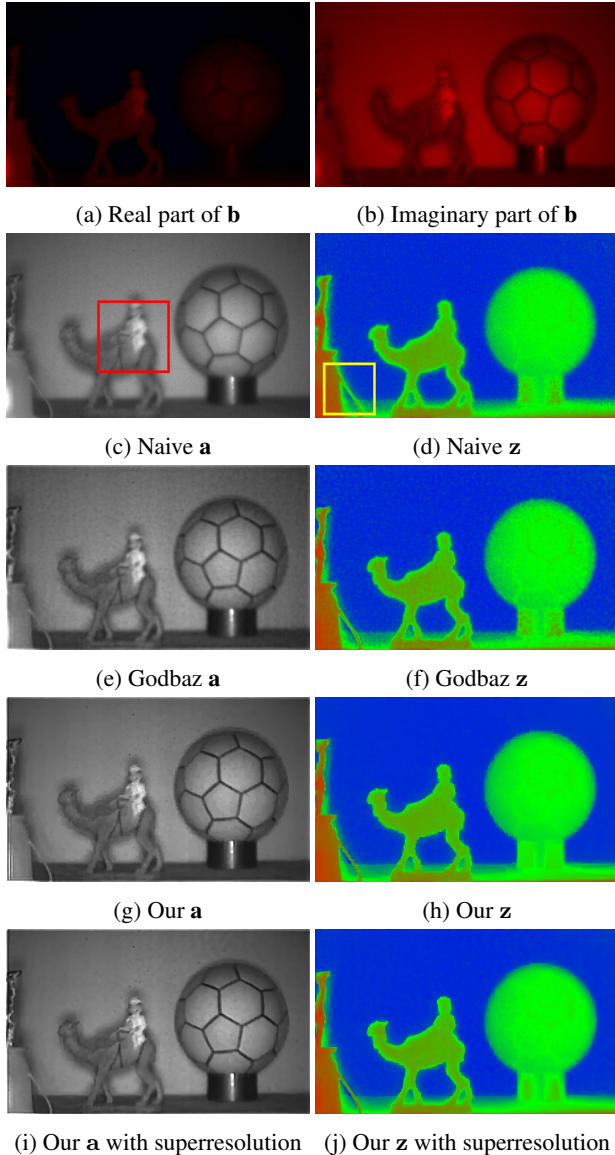


Figure 6: Results on the Camel scene. In (a-b) the red color indicates positive values and blue negative in the raw measurements. The cropped regions are shown in Fig. 7.

We run our highly unoptimized Matlab code with a single core on an Intel i7 2.4GHz CPU. The running time is reported taken the Character scene as an example. During the total 10 iterations in Algo. 1 with no superresolution, our code took 80 seconds for updating the slack variable  $\mathbf{c}$  (Sec. 3.3), 16 seconds for the amplitude (Sec. 3.4), and 347 seconds for the depth (Sec. 3.5). We believe the code can be further accelerated by choosing more efficient solvers for some subproblems or running on GPU. For example, the current Levenberg-Marquardt Matlab solver used in Algo. 3 can be replaced with much more efficient ones [1].

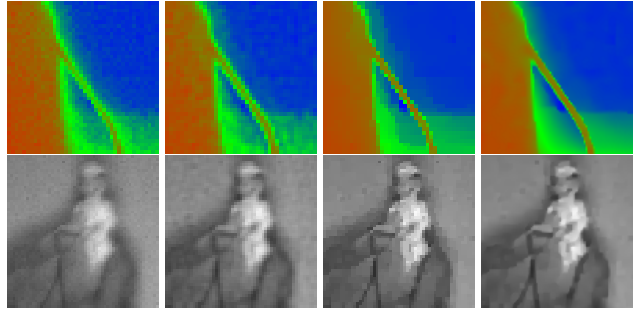


Figure 7: Two insets of the results on the Camel scene in Fig. 6. From left to right shows the naive method, Godbaz *et al.* method, ours, and ours with superresolution.

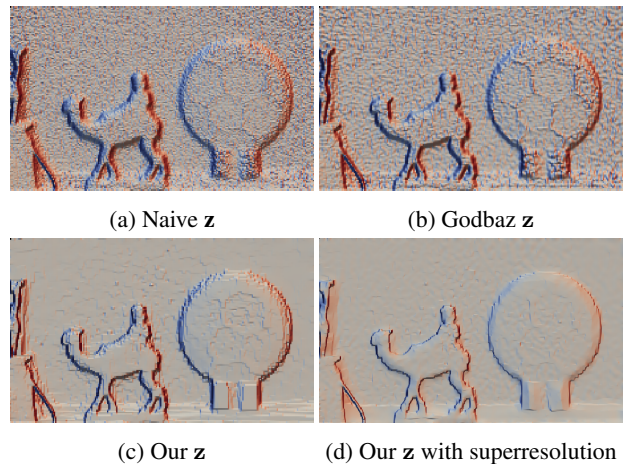


Figure 8: Mesh visualization for the Camel scene in Fig. 6. The color indicates the surface normal in horizontal direction, i.e., blue indicates left-faced surface and red the opposite.

## 6. Discussion

**Regularization strategy.** To verify the benefit of directly regularizing and solving the latent amplitude and depth, we replaced the Gaussian derivative prior in Godbaz *et al.* [9] with the TGV<sup>2</sup> prior on the complex image, and solved the complex image using ADMM framework. The result on the Camel scene is shown in Fig. 14. Even though the prior weight is high enough to over-smooth the amplitude, the estimated depth still contains strong noise compared to Fig. 6. This demonstrates the high quality of our results is to a large part owed to the approach of regularizing  $\mathbf{z}$  and  $\mathbf{a}$  directly.

**Joint deblurring and superresolution.** To show the advantage of our *joint* deblurring and superresolution from ToF raw measurements, we compare with the results of superresolution *after* deblurred by each method. Two example insets are shown in Fig. 15. Our jointly deblurred and

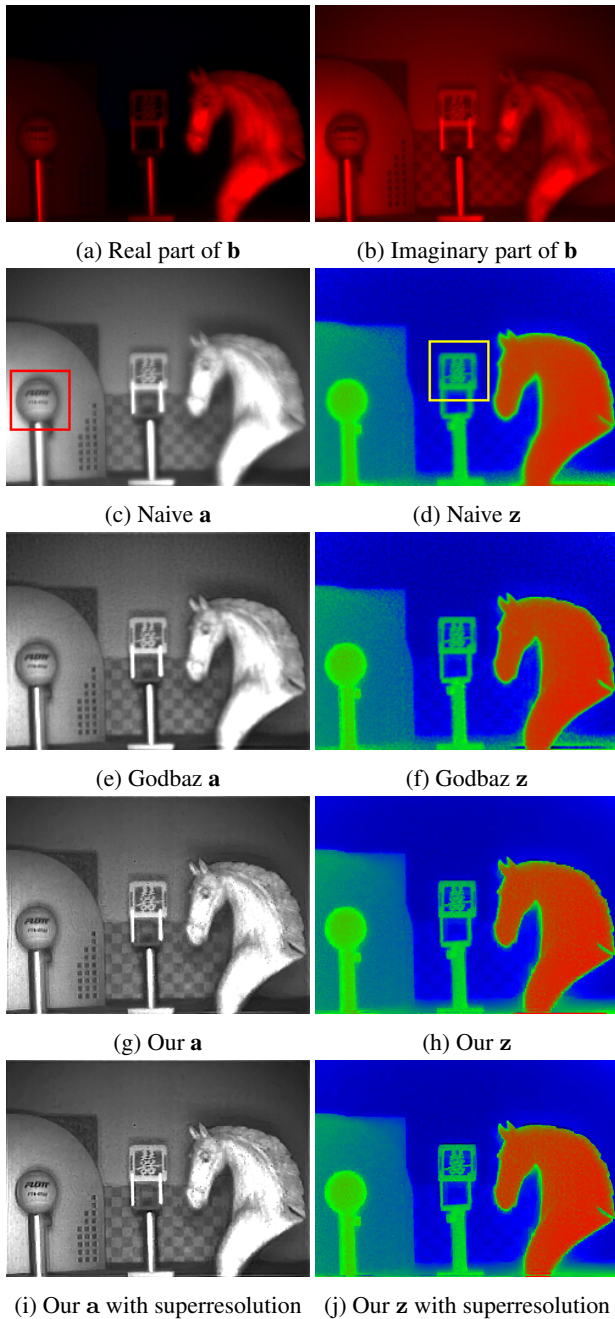


Figure 9: Results on the Character scene. The cropped regions are shown in Fig. 10.

super-resolved depth and amplitude preserve sharp features and reduce flying pixels better than the others.

**Multiple frequencies, phases and exposures.** The latest generation of ToF cameras uses multiple frequencies or phases in order to reduce range ambiguity and improve depth resolution. Multiple exposures could be used to increase the dynamic range of the raw measurements and re-

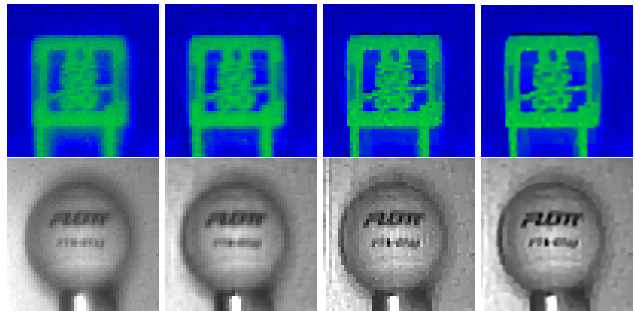


Figure 10: Two insets of the results on the Character scene in Fig. 9. From left to right shows the naive method, Godbaz *et al.* method, ours, and ours with superresolution.

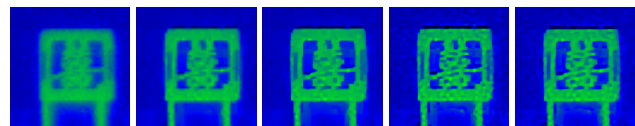


Figure 11: We run the proposed algorithm on an inset of the Character scene with different upsampling ratios. From left to right shows the result of naive method, and ours with 1x, 2x, 4x, 6x superresolution respectively. We observed that little more details are recovered beyond 4x upsampling.

move artifacts due to lack of reflection or over-saturation in one shot. The proposed algorithm well adapts for these cameras since the latent amplitude and depth is solved directly from the raw measurements, which could come from different captures and put together in the data-fitting term.

**Defocus level.** The pixel width of current ToF sensor is approximately  $45\mu\text{m}$ , compared with RGB sensors which have approximately  $5\mu\text{m}$  pixel size. As the ToF sensor resolution increases as the technology matures, the defocus effect in ToF imaging is expected to be more obvious and the importance of deblurring ToF images will become more pronounced.

**Limitations and future work.** Both the proposed method and the compared Godbaz *et al* [9] assume white Gaussian noise in the raw measurements. This noise model is inaccurate due to the non-linear image formation model, typically relatively low light conditions, as well as ambient light canceling in ToF imaging [10], which amplifies shot noise. As a future work we would like to study more accurate noise models for ToF cameras.

## 7. Conclusion

In this paper we proposed an effective method to simultaneously remove lens blur and increase image resolution for ToF depth cameras. Our algorithm solves the latent amplitude and depth directly from the raw complex images, and

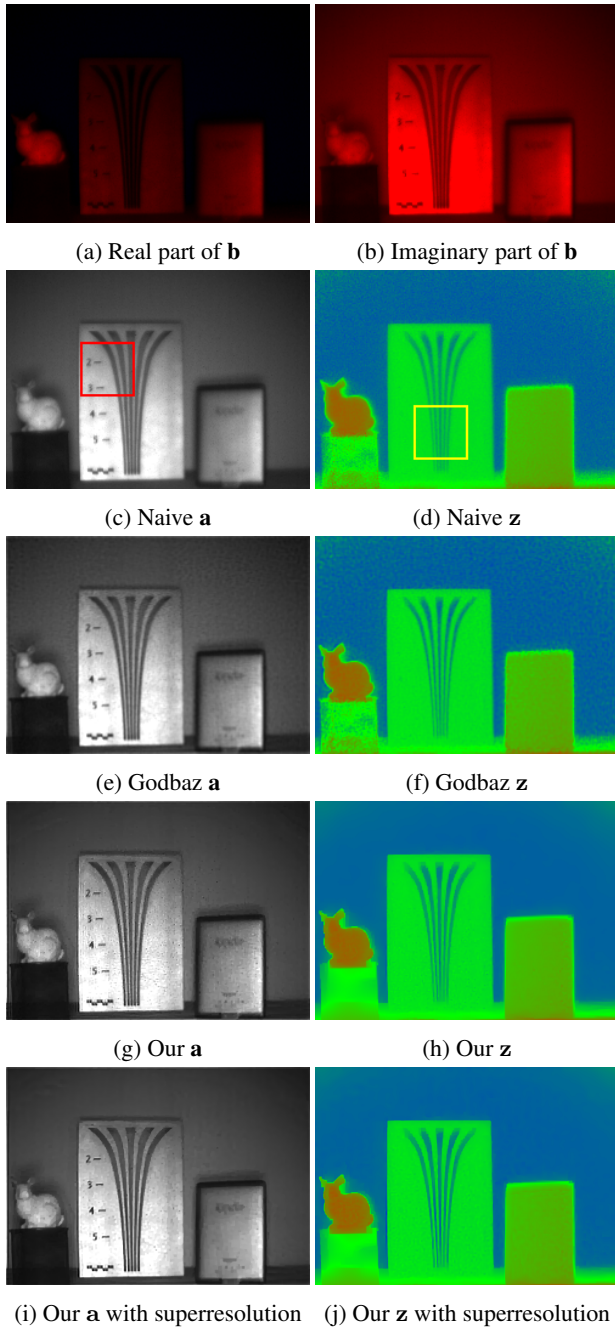


Figure 12: Results on the Board scene. The cropped regions are shown in Fig. 13.

separate priors are used for each to recover sharp features and reduce flying pixels and noise. We show our algorithm significantly improves the image quality on simulated and real dataset compared with previous work. Unlike previous approaches, our method is not fundamentally limited to the cosine model for continuous-wave ToF cameras, which has been shown to be inaccurate for many systems (e.g. [12])

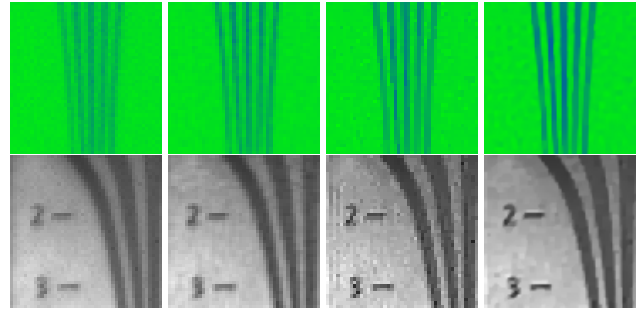


Figure 13: Two insets of the results on the Board scene in Fig. 12. From left to right shows the naive method, Godbaz *et al.* method, ours, and ours with superresolution.

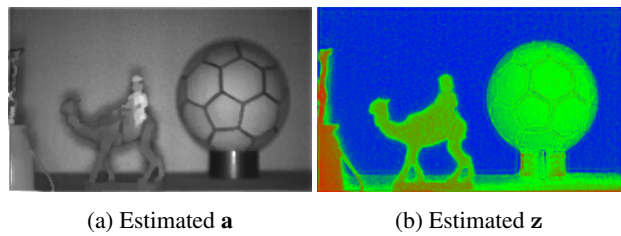


Figure 14: Experiment results using the  $TGV^2$  prior on the complex images. The estimated amplitude is over-smoothed while the depth is still highly noisy.

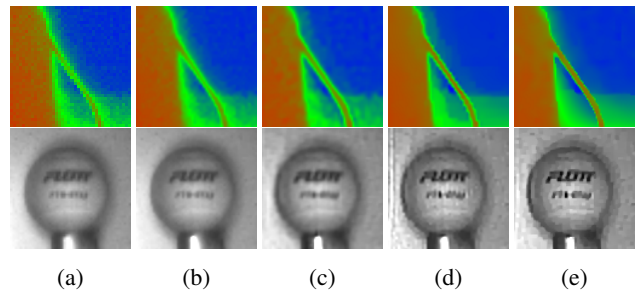


Figure 15: The top row shows depth results, and bottom amplitude. From left to right: (a) naive method; (b) naive method + post-superresolution; (c) Godbaz *et al.* + post-superresolution; (d) our deblurred + post-superresolution; and (e) our jointly deblurred and superresolved. The  $TGV^2$  prior is used for all post-superresolution. These two scenes are from Fig. 7 and 10.

and should adapt to multi-frequency, multi-phase or multi-exposure ToF cameras.

## Acknowledgements

This work was supported by Baseline Funding of the King Abdullah University of Science and Technology, an NSERC Discovery Grant, and a UBC 4 Year Fellowship.



## References

- [1] S. Agarwal, K. Mierle, and Others. Ceres solver. <http://ceres-solver.org>.
- [2] H. Avron, A. Sharf, C. Greif, and D. Cohen-Or. 1-sparse reconstruction of sharp point set surfaces. *ACM Transactions on Graphics (TOG)*, 29(5):135, 2010.
- [3] J. Balzer and S. Soatto. Second-order shape optimization for geometric inverse problems in vision. *arXiv preprint arXiv:1311.2626*, 2013.
- [4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.
- [5] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.
- [6] D. Ferstl, C. Reinbacher, R. Ranftl, M. Ruether, and H. Bischof. Image guided depth upsampling using anisotropic total generalized variation. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 993–1000, Dec 2013.
- [7] D. Freedman, Y. Smolin, E. Krupka, I. Leichter, and M. Schmidt. SRA: Fast removal of general multipath for ToF sensors. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Computer Vision ECCV 2014*, volume 8689 of *Lecture Notes in Computer Science*, pages 234–249. Springer International Publishing, 2014.
- [8] J. P. Godbaz, M. J. Cree, and A. A. Dorrington. Blind deconvolution of depth-of-field limited full-field lidar data by determination of focal parameters. In *IS&T/SPIE Electronic Imaging*, pages 75330B–75330B. International Society for Optics and Photonics, 2010.
- [9] J. P. Godbaz, M. J. Cree, and A. A. Dorrington. Extending amcw lidar depth-of-field using a coded aperture. In *ACCV 2010*, pages 397–409. Springer, 2011.
- [10] S. B. Gokturk, H. Yalcin, and C. Bamji. A time-of-flight depth sensor-system description, issues and solutions. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*, pages 35–35. IEEE, 2004.
- [11] U. Hahne and M. Alexa. Exposure fusion for time-of-flight imaging. In *Computer Graphics Forum*, volume 30, pages 1887–1894. Wiley Online Library, 2011.
- [12] F. Heide, M. B. Hullin, J. Gregson, and W. Heidrich. Low-budget transient imaging using photonic mixer devices. *ACM Transactions on Graphics (TOG)*, 32(4):45, 2013.
- [13] F. Heide, M. Rouf, M. B. Hullin, B. Labitzke, W. Heidrich, and A. Kolb. High-quality computational imaging through simple lenses. *ACM Transactions on Graphics (TOG)*, 32(5):149, 2013.
- [14] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568. ACM, 2011.
- [15] M. Kiechle, S. Hawe, and M. Kleinsteuber. A joint intensity and depth co-sparse analysis model for depth map super-resolution. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 1545–1552. IEEE, 2013.
- [16] A. Kirmani, A. Benedetti, and P. A. Chou. Spumic: Simultaneous phase unwrapping and multipath interference cancellation in time-of-flight cameras using spectral methods. In *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, pages 1–6. IEEE, 2013.
- [17] F. Knoll, K. Bredies, T. Pock, and R. Stollberger. Second order total generalized variation (tgv) for mri. *Magnetic resonance in medicine*, 65(2):480–491, 2011.
- [18] K. Levenberg. A method for the solution of certain nonlinear problems in least squares. *Quarterly of Applied Mathematics*, 2:164–168, 1944.
- [19] A. Levin, R. Fergus, F. Durand, and W. T. Freeman. Image and depth from a conventional camera with a coded aperture. In *ACM Transactions on Graphics (TOG)*, volume 26, page 70. ACM, 2007.
- [20] J. Nocedal and S. J. Wright. Numerical optimization 2nd. 2006.
- [21] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. Kweon. High quality depth map upsampling for 3d-tof cameras. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1623–1630. IEEE, 2011.
- [22] S. Schuon, C. Theobalt, J. Davis, and S. Thrun. Lidarboost: Depth superresolution for ToF 3D shape scanning. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 343–350. IEEE, 2009.
- [23] R. Wang, Z. Yang, L. Liu, J. Deng, and F. Chen. Decoupling noise and features via weighted 1-analysis compressed sensing. *ACM Transactions on Graphics (TOG)*, 33(2):18, 2014.
- [24] J. Zhu, L. Wang, R. Yang, J. E. Davis, and Z. Pan. Reliability fusion of time-of-flight depth and stereo geometry for high quality depth maps. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(7):1400–1414, 2011.