

Adaptive fovea for scanning depth sensors

Zaid Tasneem¹ , Charuvahan Adhivarahan², Dingkang Wang³,
Huikai Xie³, Karthik Dantu² and Sanjeev J Koppal¹

The International Journal of
Robotics Research
2020, Vol. 39(7) 837–855
© The Author(s) 2020
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/0278364920920931
journals.sagepub.com/home/ijr



Abstract

Depth sensors have been used extensively for perception in robotics. Typically these sensors have a fixed angular resolution and field of view (FOV). This is in contrast to human perception, which involves foveating: scanning with the eyes' highest angular resolution over regions of interest (ROIs). We build a scanning depth sensor that can control its angular resolution over the FOV. This opens up new directions for robotics research, because many algorithms in localization, mapping, exploration, and manipulation make implicit assumptions about the fixed resolution of a depth sensor, impacting latency, energy efficiency, and accuracy. Our algorithms increase resolution in ROIs either through deconvolutions or intelligent sample distribution across the FOV. The areas of high resolution in the sensor FOV act as artificial fovea and we adaptively vary the fovea locations to maximize a well-known information theoretic measure. We demonstrate novel applications such as adaptive time-of-flight (TOF) sensing, LiDAR zoom, gradient-based LiDAR sensing, and energy-efficient LiDAR scanning. As a proof of concept, we mount the sensor on a ground robot platform, showing how to reduce robot motion to obtain a desired scanning resolution. We also present a ROS wrapper for active simulation for our novel sensor in Gazebo. Finally, we provide extensive empirical analysis of all our algorithms, demonstrating trade-offs between time, resolution and stand-off distance.

Keywords

LiDAR, MEMS, range sensing

1. Introduction

Depth sensors such as LiDARs, ultrasound rangefinders, and others have been used for perception in robotics (Bailey and Durrant-Whyte, 2006; Durrant-Whyte and Bailey, 2006), computer vision, augmented reality (Izadi et al., 2011), and other applications over the last two decades. They have been empirically determined to complement RGB cameras well for spatial perception in both indoor and outdoor scenes. Recently, a new wave of time-of-flight (TOF) depth sensors have transformed robot perception. These sensors modulate scene illumination and extract depth from time-related features in the reflected radiance, such as phase change or temporal delays. Autonomous cars, drone surveillance, service robots, and wearable devices have been influenced by commercially available TOF sensors such as the Microsoft Kinect (Izadi et al., 2011) and the Velodyne Puck (Halterman and Bruch, 2010).

This article describes a new scanning TOF sensor design, which allows for novel adaptive angular resolution in depth sensing. Our work is broadly related to classical approaches for camera parameter control for perception, such as active vision (Bajcsy, 1988; Bandopadhyay et al., 1988) and active illumination (Will and Pennington, 1972).

However, our sensor can quickly vary angular resolution in a manner tailored over different parts of the field of view (FOV) and has the potential to impact fast simultaneous localization and mapping (SLAM) robotics methods (Carlone and Karaman, 2017). Our new design is in contrast to mechanical methods (e.g., for focus, zoom, pose in pan-tilt-zoom (PTZ) cameras (Del Bimbo et al., 2010)) that are slow and affect the entire FOV.

Our sensor allows angular resolution control of depth measurements, much like eye foveation. Our design uses a single LiDAR beam reflected off a microelectromechanical system (MEMS) mirror. The voltages that control the MEMS actuators allow analog (continuous) TOF sensing angles. As a modulator, MEMS mirrors have well-known

¹FOCUS Lab, University of Florida, Gainesville, FL, USA

²Department of Computer Science and Engineering, University at Buffalo, Buffalo, NY, USA

³Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA

Corresponding author:

Sanjeev J Koppal, FOCUS Lab, University of Florida, Gainesville, FL 32611, USA.

Email: sjkoppal@ece.ufl.edu

advantages of high-speed and fast response to control (Petersen, 1980).

Our designs provide a new framework to exploit directional control for depth sensing for applications relevant to small robotic platforms. Our experiments use a pulse-based LiDAR, but because the algorithms use direct depth measurements, it is safe to assume that they can easily be extended to continuous wave systems, as well as any other method for modulation such as mechanical (Halterman and Bruch, 2010) or optoelectronic (Poulton et al., 2017).

Our contributions are as follows.

- We provide imaging strategies for directional control of TOF samples, with a particular focus on the angular support of the sensing beam. We demonstrate, through real experiments and simulations, that deblurring the measurements using the sensor's angular support can recover high-frequency edges, correct non-uniform sampling, and is robust through wide FOV distortions.
- We discuss an information-theory-based control algorithm for the MEMS mirror to decide which scan to generate, given the previous measurements. By changing the cost function in the control algorithm, we can create energy-efficient TOF 3D sensing, where the algorithm places samples where they are most needed. Our method optimizes 3D sensing accuracy along with physical constraints such as range-derived power consumption, motion of objects, and free-space coverage.
- We demonstrate all of our algorithms on a real sensor, and show additional applications that are relevant for small robotic platforms, such as LiDAR zoom, which allows the controller to investigate interesting regions in the scene, as well as gradient-based estimation, which allows a constrained system to place its samples along edges, and reconstructs the scene post-capture.
- We demonstrate the advantages of LiDAR zoom over a fixed angular resolution sensor in terms of spatial density, angular resolution, and scanning efficiency on a small robotic platform.

This article is an extension of our conference paper (Tasneem et al., 2018) published at Robotics: Science and Systems in 2018. For this submission, we add the analysis of the design parameters for our adaptive sensing algorithm, demonstrate the advantages of our zooming algorithm over legacy TOF sensors on a robotic platform, release a wrapper in Gazebo to simulate the sensor, and present a foveated sensor model that allows for efficient robot motion in SLAM algorithms.

To better view the visualizations of our results in three dimensions, please see the video on our website.¹

2. Related work

2.1. 3D range sensing

The techniques used for 3D sensing can be classified in to three categories based on the measuring principles they rely

on (Beraldin et al., 2003). (i) Disparity has been used in stereo cameras (Murray and Jennings, 1997) and structured light setups (Scharstein and Szeliski, 2003) to measure depth information. Both methods rely on finding exact correspondences and fail in cases of occlusion, camera defocus blur, etc. There have been many advances in structured light setups (Zabatani et al., 2019) to improve the frame rate and depth resolution using spatiotemporal coded projections. (ii) The focus/defocus information in an image can also be used to estimate depth, but needs multiple images and might not work for dynamic scenes. (iii) TOF, on the other hand, uses the time/phase delay between sent and received light to directly estimate depth of a scene

2.2. TOF imaging and adaptive optics

Efficient TOF reconstruction is possible in the face of global illumination by encoding phase frequencies (Gupta et al., 2015) or through efficient probing (O'Toole et al., 2014). Camera exposure synchronization has enabled reconstruction in ambient light (Achar et al., 2017). Transient imaging is possible using ultra-fast lasers (Velten et al., 2016), and has recently been demonstrated using mobile off-the-shelf devices (Heide et al., 2013). In addition, spatiotemporal encodings have been shown to be efficient for both structured light illumination (O'Toole et al., 2015) and TOF illumination (O'Toole et al., 2014). We focus on 3D reconstruction and show that directional control can allow for novel types of efficiencies in sampling and energy consumption. Finally, TOF sensors for long-range sensing through atmosphere uses fast adaptive optics to remove atmospheric turbulence effects (Beckers, 1993; Tyson, 2015), whereas we target scene-adaptive sensing for autonomous systems.

2.3. Adaptive sampling in 3D models

Adaptive sampling techniques from signal processing (Principe et al., 2000) are used extensively for efficient mesh representations of computer-generated scenes (Campbell and Fussell, 1990; Terzopoulos and Vasilescu, 1991). In robotics and vision, information theoretic approaches are used to model adaptive 3D sensing for SLAM and other applications (Charrow et al., 2015; Hollinger and Sukhatme, 2014; Thrun et al., 2005). Although, there has been work in the robotics and vision community on adaptive sensing of features in the scene relevant to a particular inference task, they do not incorporate the working principles of a particular 3D sensor in their algorithm design (Denzler and Brown, 2002; Paletta et al., 2000). In the field of structured light, Zhang et al. (2014) and Rosman et al. (2016) incorporate the number of projected patterns as a resource expenditure which they try to minimize while maximizing the information gain from the scene. In this article, we are interested in adaptive algorithms for LiDAR sensors that take into account physical constraints such as the power expended on far away objects

or on objects moving out of the FOV. We demonstrate the balancing of such efficiency goals with 3D reconstruction quality.

2.4. MEMS mirrors for vision

The speed and control of MEMS mirrors have been exploited for creating imperceptible structured light for futuristic office applications (Raskar et al., 1998) and interactive-rate glasses-free 3D displays (Jones et al., 2007). MEMS mirror-modulated imaging was introduced through reverse engineering a digital light processing (DLP) projector (Nayar et al., 2006) for tasks such as edge detection and object recognition. Coupling a DLP projector with a high-speed camera allows for fast structured light and photometric stereo (Koppal et al., 2012). Adding a spatial light modulator in front of the camera allows for dual masks enabling a variety of applications (O'Toole et al., 2015), such as vision in ambient light. In contrast to these methods, we propose to use angular control to enable new types of applications for 3D imaging. We are able to play off angular, spatial and temporal sampling to allow, for example, increased sampling in regions of interest (ROIs).

2.5. Scanning LiDARs

Most commercially available LiDARs scan a fixed FOV with mechanical motors, with no directional control. MEMS modulated LiDARs have been used by NASA Goddard's GRSSLi (Flatley, 2015), ARL's Spectroscan system (Stann et al., 2014), and Innoluce, Inc. (Krastev et al., 2013). In all these cases, the MEMS mirrors are run using a fixed waveform, while we control the MEMS mirror scan path depending on scene information to demonstrate novel imaging strategies. MEMS mirror control was achieved by Kasturi et al. (2016) at Mirrorcle, Inc., who track specially placed highly reflective fiducials in the scene, for both fast 3D tracking and virtual reality (VR) applications (Milanović et al., 2011, 2017). We do not use special reflective fiducials and utilize sensing algorithms for MEMS mirror control. Finally, in Sandner et al. (2015) a MEMS mirror-modulated 3D sensor was created, with the potential for foveal sensing, but without the type of adaptive algorithms that we discuss.

3. Why adaptively control depth sensing?

Most robotic depth sensors obtain depths in a *fixed angular resolution* over the scene. For example, TOF sensors such as Pico Flexx, Kinect V2, or stereo-based sensors such as Realsense or the Kinect V1 have fixed photodetector arrays that, with a fixed lens, induce a constant set of samples of the FOV.

Instead of naive uniform sampling, it would be useful if depth sensors placed their samples where they are needed. For example, the Velodyne Puck, used in many autonomous vehicles, might benefit navigation by controlling its

mechanical motor to spend more time sampling pedestrians and cars. However, doing such adaptive sensing using slow, mechanical motors is inefficient and causes wear and tear. Instead, we build a LiDAR whose pulsed laser is reflected off a small mirror. By moving the mirror through a desired path, we can efficiently, adaptively sample the scene.

4. MEMS-modulated LiDAR imaging

A MEMS-modulated LiDAR imager has the following advantages.

- The MEMS mirror's angular motion is continuous over its FOV.
- The MEMS mirror can move selectively over angular ROIs.

In this section, we discuss some preliminaries that are needed to actualize these advantages in imaging algorithms. We first show how to use the advantage of continuous motion to remove deblurring artifacts. We then discuss how to use the advantage of selective motion to enable TOF measurements that maximize an information theoretic metric.

4.1. Sensor design and calibration

A MEMS-modulated LiDAR imager consists of a TOF engine and a MEMS modulator, as shown in Figure 1(a). The engine contains a modulated laser transmitter, a receiving photodetector that measures the return pulses, and additional electronics to calculate the time between transmitted and received pulses.

To avoid errors due to triangulation, we co-locate the centers of projection of the transmitter and receiver, as shown in Figure 1. Unlike previous efforts, such as those by Flatley (2015) and Stann et al. (2014), we do not co-locate the FOVs of the transmitter and receiver, i.e., the MEMS mirror is not our transmitter's optical aperture. This allows us to avoid expensive and heavy gradient-index (GRIN) lenses to focus the laser onto the MEMS device. Instead we use a simple, cheap, lightweight short-focus thin lens to defocus the receiver over the sensor's FOV. This introduces a directionally varying map between the time location of the returned pulses's peak, and the actual depth. We correct for this with a one-time calibration, obtained by recovering 36 measurement profiles across 5 fronto-parallel calibration planes placed at known locations, as shown in Figure 2.

4.1.1. Current configuration specs. In Figure 1(b), we show our current configuration, where we use open-source 1.35 W Lightware SF02/F LiDAR and a Mirrorcle 3.6 mm Al-coated electrostatic MEMS mirror. The LiDAR operates in near-infrared (NIR; 905 nm) and the MEMS response is broadband up to long-wave IR (14 μm). As shown in Figure 2(a), voltages from an oscilloscope control the

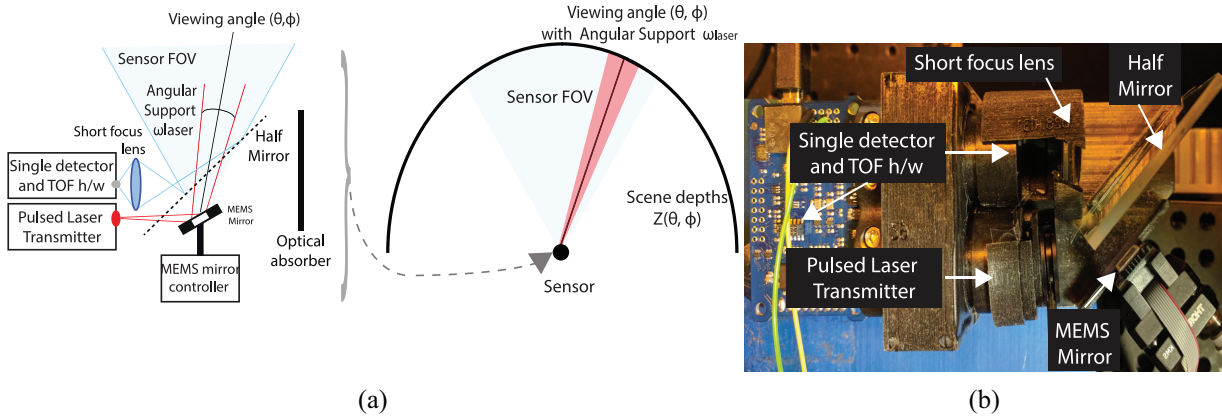


Fig. 1. (a) Co-located ray-diagram of the pulsed LiDAR modulated by a MEMS mirror and setup of our sensor: Lightware LiDAR and Mirrorcle mirror.

MEMS mirror direction, and the synchronized received pulses are inverted.

The short-focus lens introduces an interesting trade-off among FOV, range, and accuracy. At the extreme case, with no lens, our FOV reduces to a single receiving direction with the full range of the LiDAR (nearly 50 m). As we increase the FOV, and the defocus, the signal-to-noise ratio (SNR) received at the transducer decreases, reducing range. While we can compensate with increased gain, this introduces noise and reduces accuracy. In this article, we traded-off range for accuracy and FOV, and our device has a FOV of $\approx 15^\circ$, a range of 0.5 m, and is set at the lowest gain (highest SNR).

The Lightware LiDAR sampling rate is 32 Hz, which, in this article, restricts us to static scenes. We prefer this LiDAR, despite the low rate, because it allows for raw data capture, enabling design-specific calibration. We perform multiple scans of the static scene and average our measurements to improve accuracy. After the calibration in Figure 2(b) we reconstruct a plane at 27 cm (not in our calibration set) and obtained a standard deviation of 0.61 cm (i.e., almost all points are measured in a ± 1.5 cm error range), as shown in Figure 2(c).

The total weight of our system is approximately 500 g; however, most of that weight (~ 350 g) is contained in a general-purpose oscilloscope and MEMS controller, and it would be trivial to replace these with simple, dedicated circuits (67 g Lidar, 187 g oscilloscope, 74 g enclosure, 10 g optics, and 147 g MEMS controller).

4.2. Directional control of TOF sensing

Voltages over the MEMS device's range physically shift the mirror position to a desired angle, allowing for range sensing over the direction corresponding to this angle. Let the function controlling the azimuth be $\phi(V(t))$ and the function controlling elevation be $\theta(V(t))$, where V is the input voltage that varies with time t . Without loss of generality, we assume a pulse-based system, and let the firing rate of the LiDAR/TOF engine be $\frac{1}{T_f}$ Hz, or T_f seconds

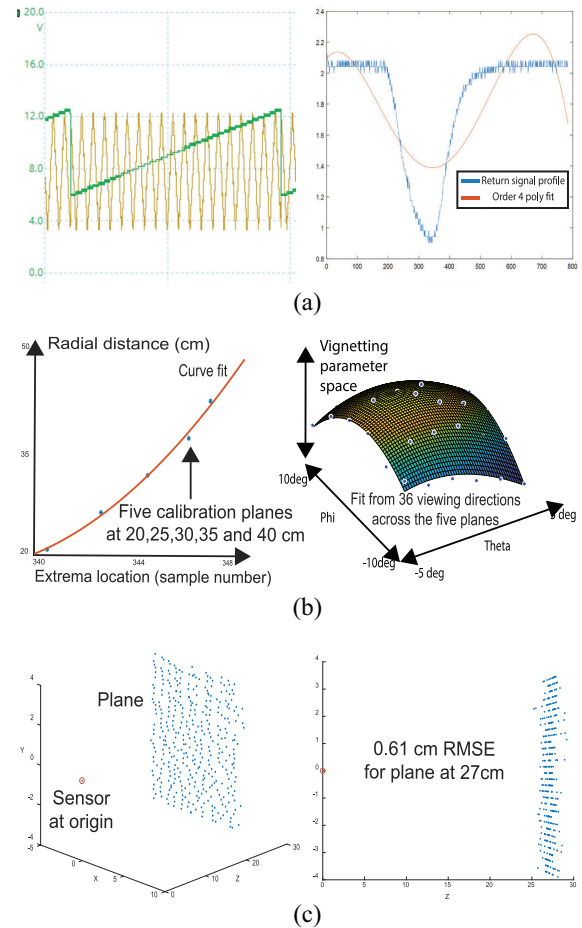


Fig. 2. Calibration. In (a), we show a screenshot of the voltages that control the MEMS mirror pose, as well as an example of the return pulse. We fit a fourth-order polynomial to the return pulse to detect the extrema location. In (b), we show how to map this extrema location to depth in centimeters, by collecting data across five planes at known depths. This calibration also involves a vignetting step, to remove effects in the receiver optics. In (c), we validate our sensor by reconstructing a fronto-parallel plane at 27 cm, showing a standard deviation of 0.61 cm (i.e., almost all points are measured in a ± 1.5 cm error range). In the current configuration, the FOV is $\approx 15^\circ$ and the range is 0.5 m.

between each pulse. For continuous wave-based systems $\frac{1}{T_f}$ Hz is equivalent to the frequency of emission. Therefore, the n th measurement of the sensor happens along the ray direction given by the angles $(\theta(V(n T_f)), \phi(V(n T_f)))$.

The world around a miniature vision sensor can be modeled as a hemisphere of directions (Figure 1(a) center), i.e., the plenoptic function around the sensor is an environment map parameterized by the azimuth and elevation angles. Conventional imaging systems are characterized by their point-spread function (PSF) (Goodman, 2005) whereas miniature vision systems are characterized by their angular support ω (standard symbol) (Koppal et al., 2013). For miniature active scanning TOF systems, the angular spread of the laser beam determines the angular support, which we term as ω_{laser} in Figure 1(a).

4.3. Correlated depth measurements over the FOV

Each sensor measurement occurs across the laser's dot size, given by the angular support ω_{laser} . Let us now define the separation *between* measurements in angular terms, as ω_{diff} . For many commercial LiDARs, such as the Velodyne HDL-32E, the measurement directions are further apart than the angular support, i.e., $\omega_{\text{diff}} \geq \omega_{\text{laser}}$.

For our system, the measurement separation ω_{diff} depends on the differential azimuth elevation, given by $\omega_{\text{diff}} = \delta\phi \delta\theta \sin(\phi)$, where ϕ and θ were defined previously. MEMS modulation allows almost any angle inside the sensor's FOV. Therefore, if the measurements satisfy the inequality $\omega_{\text{diff}} \leq \omega_{\text{laser}}$, then the measurements are correlated.

4.4. Linear filtering with MEMS mirrors

The correlation-enabling inequality defined above allows us to think about our MEMS modulated sensor as similar in properties to fixed-resolution imagers. In particular, if the inequality is satisfied, then the measured LiDAR depth measurements y are some linear transform \mathbf{B} of the "ground truth" LiDAR measurements defined at some canonical, desired resolution as \mathbf{z}

$$y = \mathbf{B}z \quad (1)$$

where, crucially, \mathbf{B} can be full rank only if the inequality $\omega_{\text{diff}} \leq \omega_{\text{laser}}$ is true. Many conventional LiDAR systems have large gaps between measurements, implying that \mathbf{B} has many zero rows and the inequality is not held. On the other hand, if this inequality is satisfied, then the rich body of work in vision using controllable PSFs can be applied here, including image deblurring (Raskar et al., 2006), refocussing (Ng, 2005), depth sensing (Levin et al., 2007), and compressive sensing (Fergus et al., 2006). We now provide some examples, augmented by illustrations and noisy simulations in Figure 3 of a 2D toy scene where a MEMS modulated LiDAR is shown to be scanning a circular scene with a sharp discontinuity. In these examples, the rows of

the measurement matrix are the indices of different measurements, and the columns cover discrete viewing directions across the FOV. Assuming uniform laser dot intensity, we can represent the angular support for any particular MEMS mirror position as an indicator vector along viewing direction. Any measurement collects information across the angular support in this FOV, given as white, and ignores the rest, shown as black.

4.4.1. Uniform sampling and fixed filter. In Figure 3(a), the angular support ω_{laser} is larger than the angle offset ω_{diff} , and therefore the direct measurements blur the high-frequency information in the sharp discontinuity in Figure 3(a)(center left). The measured, received pulses at the sensor are, given the measurement equation mentioned above, $y = \mathbf{B}z$, and recovering the ideal depths \mathbf{z} is a deblurring problem. In the case of uniform motion, the angular support of the sensor is constant across viewing direction, because it is simply the angular spread of the laser being reflected off the MEMS mirror. The uniform sampling and constant filter size (angular support) result in measurements offset by the same step, creating a near-perfect banded diagonal matrix \mathbf{B} in Figure 3(a)(center right), which is invertible. We apply non-negative least squares to obtain measurements as shown in Figure 3(a)(right), with zero root-mean-squared error (RMSE).

4.4.2. Non-uniform sampling and fixed filter. We note that the angular support can be affected by adding laser optics, as shown in Figure 3(b), where the angular support ω_{laser} is increased. This would be necessary if the maximum angular spread between measurements is much larger than the original angular support ω_{laser} , due to non-uniform MEMS mirror control. In fact, such control occurs naturally with MEMS devices driven by linear signals, since the MEMS device's forces follow Hooke's law of springs (Yang et al., 2013). In Figure 3(b)(center right), the non-uniform and blurred measurements result in a banded matrix with varying bandwidth. The ground-truth recovered by inverting \mathbf{B} is both accurate and has the desired uniform density sampling.

4.4.3. Non-uniform sampling and variable filter. Finally, consider the effect of a wide-angle optical system, such as a fish-eye lens. This would shear the diagonal band in the matrix \mathbf{B} , where extreme angles would integrate large portions of the FOV, which samples closer to the optical axis would show finer angular resolution. The smooth motion of the MEMS mirror allows us to invert or redistribute the samples across the FOV, removing wide-angle distortion. In Figure 3(c), we show an example of such optics (Koppal et al., 2013; Yang et al., 2017), which has been used recently in wide-angle MEMS modulation. Using the equation from Koppal et al. (2013), we generate the viewing-dependent angular support that creates a blurred version of

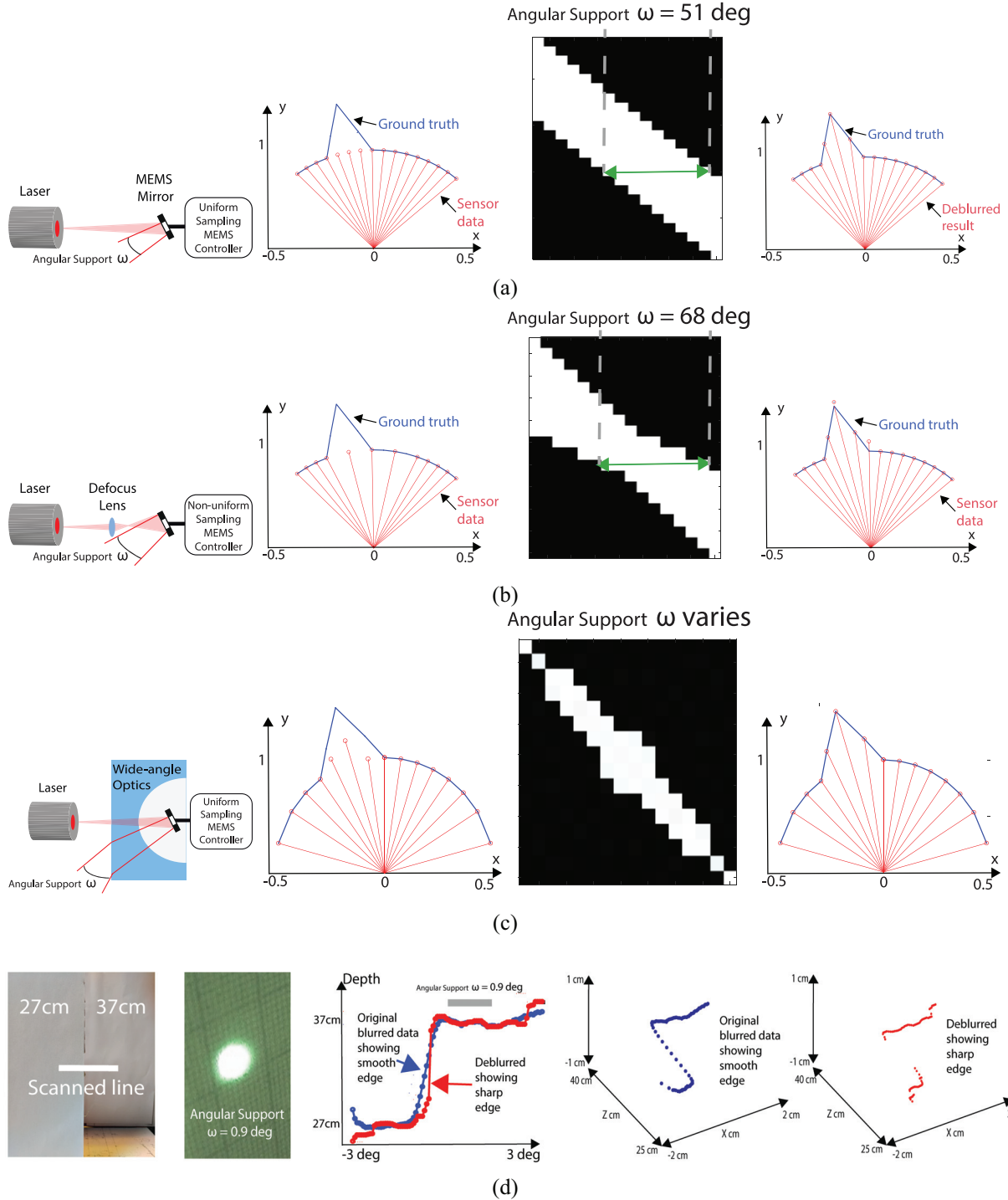


Fig. 3. Deblurring using angular support: (a) deblurring 1D simulated data with uniform sampling; (b) deblurring 1D simulated data with non-uniform sampling; (c) deblurring 1D simulated data with wide-angle optics; (d) deblurring real 3D edge data. In (a)–(c), we show simulations of pulse-based LiDAR with 2% noise on a simple 2D circular scene with a sharp protrusion. In (a), the native laser dot size blurs the scene equiangularly, resulting in a banded matrix (center right) that is invertible and the measurements are recovered (right). In (b), a larger angular support is created with additional optics, for a sensor with non-uniform angular sampling. The matrix is still invertible, and can be used to resample the scene uniformly. In (c), the effect of wide-angle optics is shown, modeled from the refractive optics in Koppal et al. (2013) and Yang et al. (2017), where deblurring is still successful. Finally, (d) shows real experiments across a depth edge. Without deblurring, measurements are hallucinated across the gap.

the scene in Figure 3(c)(center left) and a corresponding \mathbf{B} matrix in Figure 3(c)(center right). This matrix is geometrically constructed to be invertible because all its values are positive and its trace is non-zero, and allows for scene recovery in Figure 3(d).

4.4.4. Real result. In Figure 3(d) we show a real deblurring result. The scene is two planes at 27 and 30 cm, where the sensing angles follow a great arc in the hemisphere of directions, as shown by the line segment in Figure 3(d)(left). We measure the angular support ω_{laser} as 0.9° as shown in Figure 3(d)(center left). Without deblurring, the original measurements result in a smoothed edge, as shown in blue. We use the damped Richardson–Lucy deconvolution optimization algorithm that takes our measured angular support as a starting point. This results in a strong edge recovery, with fewer incorrect measurements, as shown in red in the figure.

4.5. Adaptive sensing strategy overview

Our adaptive sensing strategy is to bring active vision (i.e., path planning) into the LiDAR sensor. Instead of controlling a robot, the planning algorithm control the scan path of the MEMS mirror, which translates into a new measurement of the scene structure.

Like many planning techniques, our approach relies on first specifying local quality measures of the image. Once local weights are known, candidate global trajectories are proposed that are subject to physical constraints. In robotics, these are usually constraints on where and how fast the robot can move. For our adaptive LiDAR, the constraints are on how fast the MEMS can move and what trajectories are physically feasible.

4.6. Adaptive TOF sensing in selected ROIs

The MEMS mirror can modulate the LiDAR beam through a range of smooth trajectories. A unique characteristic of our setup is that we can adapt this motion to the current set of scene measurements. We control the MEMS mirror over the FOV by exploiting strategies used for LiDAR sensing in robotics (Charrow et al., 2015; Julian et al., 2014; Thrun et al., 2005). In particular, we first generate a series of candidate trajectories that conform to any desired global physical constraints on the sensor. We then select from these candidates by maximizing a local information theoretic measure that has had success in active vision for robotics (Charrow et al., 2015).

4.6.1. Candidate trajectories from global physical constraints. To generate a series of candidate trajectories, we encode the scene into regions where the sensing beam should spend more time collecting many measurements, and regions where the beam should move quickly, collecting fewer measurements. We achieve this by clustering the

scene into ROIs based on a desired physical metric. In the applications section, we show that different metrics can enable, for example, scanning the scene under the constraint of limited power. Similar metrics can be specified for time or scene complexity.

We first tessellate the current scene scan in three dimensions into bounding boxes $B_i(X_c, Y_c, Z_c, H)$, which contain all points (X, Y, Z) in the current scan such that these lie in a box centered at (X_c, Y_c, Z_c) with side length given by H . We require that a metric M be designed such that $M(B_i) \in \mathbf{R}$. We then apply an unsupervised clustering mechanism, such as k -means, to the set of boxes, where the feature to be clustered from each box B_i is $(M(B_i), X_c, Y_c, Z_c)$. Automatically finding the number of clusters is an open problem in pattern recognition, and while a variety of methods exist to find an optimal k , for simplicity we generate candidate trajectories over a range of cluster centers, from 2 until k_{max} , which we leave as a design parameter.

Each cluster of boxes defines a ROI in the scene. Let us describe the solid angle subtended by the ROI onto the MEMS mirror, indexed by j as ω_j , and let the weighted average physical metric of all the boxes in the j th ROI be m_j . If there are n samples across the FOV, then our goal is to create a series of voltages $V(t)$, such that the angles generated maximize the following cost function,

$$\max_{V(t)} \sum_i^{(n T_f)} \sum_j^k F(\theta(V(i)), \phi(V(i)), \omega_j, m_j) \quad (2)$$

where i varies from 1 to the number of LiDAR samples nT_f , where k is the number of ROI clusters in that scan (varying from $j=1$ to the number of clusters k), $\frac{1}{T_f}$ is the firing of the LiDAR/TOF engine and where F is a function that outputs $e^{\alpha m_k}$ if $(\theta(V(n T_f)), \phi(V(n T_f)))$ lie inside ω_k . Here α is a user-defined weight that controls the impact of the relative score of the different ROIs on the time spent in each ROI. If $m_{\text{total}} = \sum_j^k m_j$, we pick a weight α such that the time spent in each ROI is proportional to $\frac{m_j}{m_{\text{total}}}$.

Note that the above equation does not contain a derivative term $V'(t)$ to enforce smoothness, because we generate only candidate trajectories that conform to physically realizable MEMS mirror trajectories, such as sinusoids, triangular wave functions, and raster scans. We generate p such trajectories ψ_1, \dots, ψ_p from a global level algorithm detailed in Algorithm 1 and pick the scan that maximizes the sum of the information gain from each scanned ray

$$\max_{V_p(t), m} I(m|\psi_i) \quad (3)$$

where ψ_i is the current scan and m is the probabilistic occupancy map of the scene calculated by tessellating the scene into voxels $B'_i(X'_c, Y'_c, Z'_c, H')$, and where the probability of occupancy is given by $e^{-0.5r}$, where r is the radial distance between the voxel center and the nearest measured scan point. In other words, our strategy is to optimize the goal

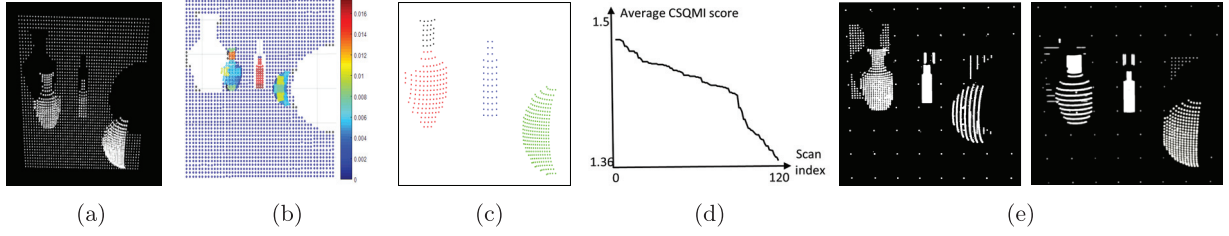


Fig. 4. Simulation of adaptive LiDAR. In (a), we show a base scan of a scene with three simple objects. This base scan is tessellated into boxes, which are scored according to some desired metric. In (b), we show a geometric score based on the residual of 2D principal component analysis of points in a box. Note the background plane has a low score. This score is used with multiple values of cluster number k to generate many ROI segmentations, as in (c). The ROIs globally direct the MEMS mirror to scan longer in regions with a higher average box score. For each ROI segmentation, many candidate scans are generated by varying scan parameters such as phase, shape, directionality, etc., whose CSQMI scores are shown in (d). In (e), we show the highest and lowest average CSQMI scores of these scans, and the highest scan's MEMS mirror motions would be the actual trajectories scanned next.

in Equation (2) by optimizing the gain within each cluster as in Equation (3).

We do this by utilizing a form for $I(m|\psi_i)$ in Equation (3) derived from the Cauchy–Schwarz quadratic mutual information (CSQMI) for a single laser beam (Charrow et al., 2015). The expression for CSQMI is reproduced here Charrow et al. (2015),

$$\begin{aligned} & \log \sum_{l=0}^C w_l \mathcal{N}(0, 2\sigma^2) + \log \prod_{i=1}^C (o_i^2 + (1 - o_i)^2) \\ & \sum_{j=0}^C \sum_{l=0}^C p(e_j) p(e_l) \mathcal{N}(\mu_l - \mu_j, 2\sigma^2) \\ & - 2 \log \sum_{j=0}^C \sum_{l=0}^C p(e_j) w_l \mathcal{N}(\mu_l - \mu_j, 2\sigma^2) \end{aligned} \quad (4)$$

where C refers to the number of “cells,” voxels intersected by the current laser ray direction, \mathcal{N} and σ define the mean and variance of a Gaussian model of the return pulse, o_i is the probability that the i th cell is occupied, $p(e_j)$ is the probability that the j th cell is the first occupied cells (with all before being unoccupied), and w_l is a weight defined by $w_l = p^2(e_l) \prod_{j=l+1}^C (o_j^2 + (1 - o_j)^2)$. As we have multiple ray directions in each candidate scan, we aggregate each of these to produce a single, overall CSQMI value for that candidate scan and pick the scan with the maximum score.

4.6.2. Simulation example. In Figure 4, we show a scene created with BlenSor (Gschwandtner et al., 2011) with three objects in front of a fronto-parallel plane. We start with an equiangular base scan shown in Figure 4(a) because all directions have uniform prior. We tessellate the scene into boxes B of size $25 \text{ cm} \times 25 \text{ cm} \times 25 \text{ cm}$ and use the residuals of a 2D principal component analysis fit to score the complexity of each box, as in Figure 4(b). Clustering the boxes, Figure 4(c) creates ROIs. Varying the number of clusters and varying scan parameters creates a variety of candidate, each of which have a CSQMI score (Figure 4(d)). We pick the best such score, as shown in

Figure 4(e), where it is contrasted with the worst such scan. Note that, in the best scan, the neck of the vase is captured in detail and the sphere is captured equally densely across θ and ϕ angles.

4.6.3. Design parameters of adaptive TOF sensing algorithm. The performance of adaptive sensing algorithms depends on certain design parameter settings. Here we present some empirical analysis of these parameters. First, we present the robustness of the algorithm to base scan resolution, which affects the reliability of the clustering component of our method.

4.6.4. Practical issues regarding clustering. Instead of generating random trajectories globally and then using CSQMI gain measure, we use a clustering technique to segment the scene into objects (ROIs) and then the CSQMI measure to decide the best physically feasible trajectory scan within these ROIs. The assumption here is that the scene is made up of some number of “interesting” objects and a background. In other words, clustering helps us decide where to focus scanning whereas the CSQMI gain measure helps us decide what trajectory to follow in these regions. Generating random trajectories would also be unfeasible for the MEMS mirror to physically reproduce. Although we use a naive k -means technique, any more advanced clustering algorithm could be used including neural-network-based unsupervised segmentation (Qi et al., 2017). If a more advanced algorithm is used, we expect the results to be even better. Of course, as with all such methods, the number of objects (i.e., k) is not known a priori and is estimated through heuristics, and in our experiments we vary the number of clusters from $k=2$ to 5 for ROI generation in the previous presented simulation where the ground truth is $k=3$. As k -means is an unsupervised and non-deterministic clustering algorithm, we take the average over 50 different random seeds for each cluster number. We then calculate the CSQMI gain for the proposed scans using our ROI generation and importance weight (α) distribution algorithm. We vary the base scan resolution from

Algorithm 1 Global-level trajectory planning algorithm**Require:** Generating candidate trajectories from base scan (X, Y, Z) **Input:** Base Scan $(X, Y, Z) \in \mathcal{R}^3$, Number of Regions of Interest $k \in \mathcal{R}$, Dimension of bounding boxes $H \in \mathcal{R}$ **Output:** Candidate trajectories

- 1: $\psi = \{\{\theta^1, \dots, \theta^{\alpha_1 + \dots + \alpha_k}, \dots, \theta^{n_{Tf}}\},$
- 2: $\{\phi^1, \dots, \phi^{\alpha_1 + \dots + \alpha_k}, \dots, \phi^{n_{Tf}}\}\}_{1, \dots, p}$ $\{p$ is the number of candidate trajectories and α_j is the number of samples in the j th ROI for adaptive scans}
- 3: Tessellate base scan into bounding boxes $B_i(X_c, Y_c, Z_c, H)$
- 4: Calculate complexity score at each box c_j and other physical metrics
- 5: Cluster the boxes into ROIs ω_j based on (X_c, Y_c, Z_c, m_j)
- 6: **Initialization** $\psi = \{\{\}, \{\}\}$
- 7: **for** $j \leftarrow 1$ to k **do**
- 8: Calculate the average of different physical metrics over all the boxes in j th ROI
- 9: Calculate a weighted average physical metric m_j
- 10: Calculate $\alpha_j = \frac{m_j \times n_{Tf}}{m_{\text{total}}}$ {where $m_{\text{total}} = \sum_j m_j$ }
- 11: Generate q probable raster scans π_j within the ROI, $\pi_j = \{\{\theta^1, \dots, \theta^{\alpha_j}\}, \{\phi^1, \dots, \phi^{\alpha_j}\}\}_{1, \dots, q}$ {probable scans are generated by varying resolution in θ and ϕ }
- 12: Append π_j to ψ
- 13: **end for**

20×20 to 60×60 and repeat the same procedure plotting the CSQMI values against the base scan resolution, as shown in Figure 5(a). A good adaptive scanning strategy would be the one that leads to the maximum information gain in the subsequent scans.

From this figure, we can infer that as we increase the base scan density the proposed scan CSQMI gain decreases, which indicates that the adaptive TOF sensing algorithm is advantageous in gaining more information when the base scan density is low, whereas that advantage becomes less prominent for high base scan densities. In addition, when the number of clusters is precisely the ground truth $k = 3$, the algorithm closely follows the expected monotonously decreasing trend. However, this fails when k is wrong, demonstrating clustering failures across the different trials. In this analysis, the number of samples in the proposed scans is equal to the base scan resolution.

Next, we ask the question whether it is possible to save scan time and energy in consequent scans by reducing the number of samples, i.e., once the importance of the tessellated boxes has been obtained, how do the increased measurements affect CSQMI. To do this empirically, we fix the scan resolution at 50×50 and segment the base scan into the ground truth $k = 3$ clusters. In Figure 5(b), we vary the number of points in the consecutive scan from 1,000 to 2,500 (i.e., 0.4–1 times the samples in base scan) and observe that even though the information gain is proportional to the number of points in the consecutive scan, the scale of change is not large enough and, hence, we can save samples based on the application such as performing consequent scans at a low resolution while being fairly robust.

5. Applications

As described earlier, our TOF sensor has wide-ranging applications in robotics, computer vision, augmented reality (AR), and other domains. To demonstrate this, we implement a few novel applications using our sensor.

5.1. Smart LiDAR zoom

Optical zoom with a conventional fixed array of detectors involves changing the FOV so that the measurements are closer together in the angular domain. Intelligent zoom exists for conventional cameras using PTZ transformations (Deselaers et al., 2008) and light-fields (Badki et al., 2017). Here we demonstrate, for the first time, intelligent LiDAR zoom.

Suppose we are provided with angular supports of n interesting ROIs in the scene $(\omega_{\text{zoom}}^1, \omega_{\text{zoom}}^2, \dots, \omega_{\text{zoom}}^n)$ and a corresponding series of importance weights (w_1, w_2, \dots, w_n) . These could come from another algorithm, say face detection, or from a user giving high-level commands to the system.

We can use these regions and weights to modify the default effect of the adaptive LiDAR framework described in Section 4.6. For example, if a box is contained in ω_{zoom}^i , then we can increase the geometric score in the boxes by a factor determined by the corresponding importance weight w_i . This would increase the amount of time that the sensor spends in the angular support corresponding to the box.

Smart LiDAR zoom has a clear advantage over naive zoom, which would place all LiDAR samples exclusively in ROIs. This because any zoom interface must also offer scrolling. As is known in computer graphics (Nehab et al., 2006), efficient scrolling requires caching motion and data near user viewpoints, to allow for fast rendering for real-time interaction.

In Figure 6(a), we show a scene with a 3D printed flower and a bottle. We show a base scan of the scene in Figure 6(b) with equiangular samples. The user places a zoom ROI around the flower. We show that naive zoom directs the measurements entirely on the flower, with almost zero measurements around the zoomed-in area.

Although we have not implemented real-time scrolling, in Figure 6(c)–(g) we simulate the effect of scroll in naive zoom, showing a jarring transition period in the image,

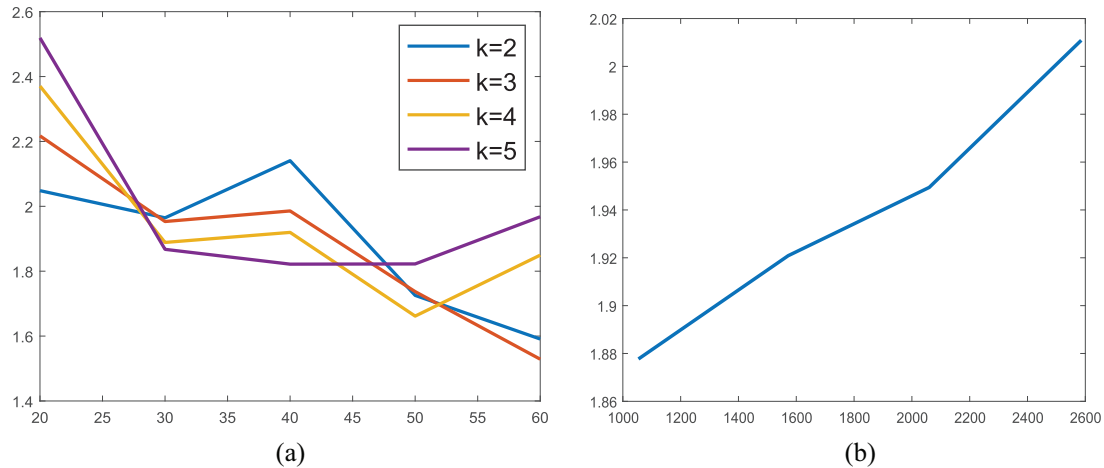


Fig. 5. Adaptive TOF sensing algorithm's design parameters. We take the simple scene from the simulation example created with Blensor, as shown in Figure 4(a), and vary the adaptive TOF sensing algorithm's parameters. In (a), we vary the angular resolution in base scan and the number of clusters in segmentation and observe the CSQMI values of the proposed scans. Our algorithm shows a clear advantage of retrieving more information from the scan if the base scan density is low. In (b), we vary the number of samples in the next scan for the base scan angular resolution of 50×50 and observe that we retrieve more information for more samples in the next scan but overall the scale of variation is not considerable.

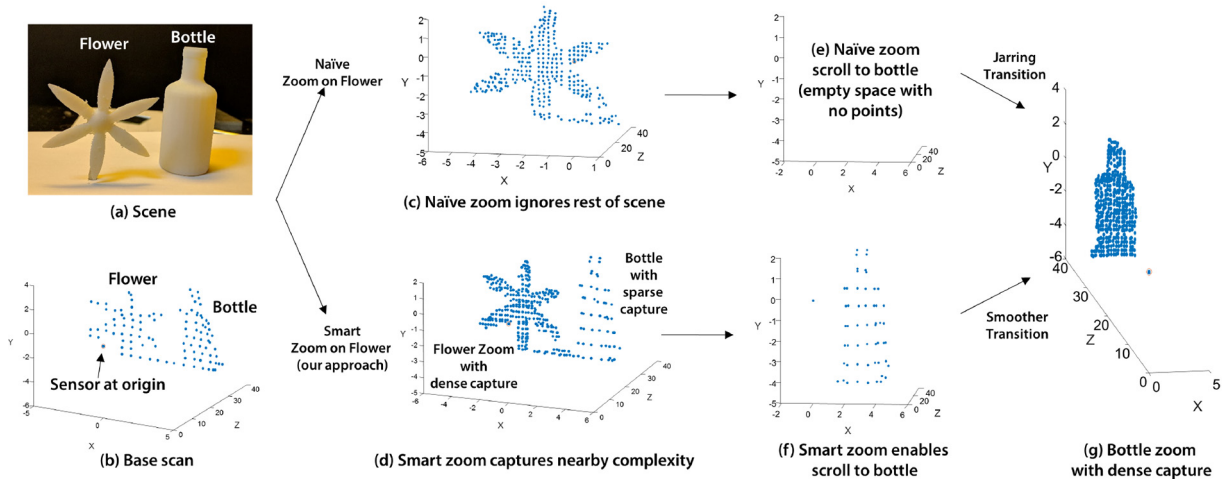


Fig. 6. Our smart LiDAR zoom versus naive LiDAR zoom. By moving the MEMS mirror to certain ROIs, we can “zoom” or capture more angular resolution in that desired region. In (a), we show a scene with two objects, and in (b), we show the output of our sensor with equiangular sampling. If the zoom shifts to the flower, then the naive zoom concentrates the samples in the base scan on the flower exclusively, in (c). On the other hand, our smart zoom (d) takes measurements outside the zoom region, depending on neighboring object's complexity and proximity. A naive approach does not visually support scrolling, because other areas of the scene are blank (e). Our smart zoom allows for scrolling to nearby objects that have some measurements (f). This allows for a smoother transition when the zoom shifts to that object (g), compared with naive zoom.

because the measurements suddenly appear in a previously blank image. Instead, our method spreads the samples across the two complex objects in the scene, allowing for a more meaningful transition when scrolling is simulated in Figure 6(g) to the dense scan when the bottle is zoomed. Note, that while the scroll motion is simulated, all the zoom measurements are real measurements from our sensor performing directionally varying sampling, based on the desired zoom area.

5.2. Zooming for mobile robotic applications

Our sensor and algorithms provide the ability to zoom into ROIs in the FOV and enable a new set of more efficient robot algorithms for various applications. In this section, we perform simple experiments with a Turtlebot² to demonstrate this potential. Our experimental setup is shown in Figure 7(a). Our sensor is used on the Turtlebot in three different modes.

- **Mode A:** We keep the angular resolution fixed at different distances, simulating the fixed resolution capability legacy TOF sensors such as Velodyne Puck and Kinect provide. Scan time and scan pattern are the same, and the robot is moving forward. In the various graphs of Figure 7(c) and (d), this is shown as a solid black line.
- **Mode B:** This mode demonstrates our zoom capability in one possible way. We keep the spatial scan density the same by adjusting the angular resolution. The scan pattern changes as the solid angle subtended by the object at the sensor origin increases as the robot moves closer. We have to readjust the zoom extents at every step. In the various graphs of Figure 7(c) and (d), this is shown as a solid red line.
- **Mode C:** This mode demonstrates our zoom capability in another possible way. We zoom in at the farthest distance, record the angular resolution in this configuration, and keep the same angular resolution as the robot moves closer to the object while the zoom extent increases. This, in turn, increases the scan time considerably. In the various graphs of Figure 7(c) and (d), this is shown as a solid blue line.

In the section on adaptive TOF sensing above, we use global physical constraints to decide the ROI and the amount of zoom given a base scan. Within these ROIs we choose the best scanning trajectory using a CSQMI measure locally. In this section though, scanning in mode A is done to give a fair comparison with fixed resolution legacy sensors and has no adaptive component. Modes B and C, on the other hand, use the guiding rule of fixed spatial scan density and fixed angular resolution, respectively, to decide the scan density after adaptive ROI prediction at every distance through clustering on mode A scan. We perform each scan independently of past information to show variation in modes and distance from scene. Locally, we keep the scan pattern fixed to a raster scan instead of varying the candidate trajectories and selecting the best one using a CSQMI measure for simplicity of comparison between modes and not within modes.

5.2.1. Comparison metrics. In order to quantitatively analyze the advantages of the different modes over each other, we define several metrics. *Spatial density* is the most important as the final output of our sensor is a point cloud and this metric represents the number of points per square centimeter of the object we were able to scan. *Scanning time* is the time spent per scan frame and is directly proportional to the total number of samples the sensor captured. This is important because all LiDAR engines have a given frequency at which they can sample points and having a low scan time can increase the frame rate, which is desirable for most applications. *Scanning efficiency* is the number of samples the sensor was able to capture in a scan that came from the object and not empty space. This represents

the percentage of the scene that was actually of interest and was captured by our sensor.

5.2.2. Line sensor simulation. We first perform a simulation to predict the consequences of different modes on these metrics. We do the analysis within a plane parallel to the ground plane and passing through the sensor origin as shown in Figure 7(a) (right) so the sensor can be seen assumed as a line sensor with our FOV and zoom capabilities capturing an object of width 20 cm perpendicular to the scanning direction and the robot is continuously moving towards the object of interest and scanning from a range of 10 to 100 cm. From Figure 7(b), we hypothesize that the spatial density for mode C is the highest, but has the trade-off of having high scan time for objects at close distances. Mode B, on the other hand, allows for constant spatial density throughout scanning while keeping the scan time constant. Because mode A cannot readjust its scan pattern, its spatial density is the worst. In addition, it is important to note that even though mode A can achieve the same scan density at close distances without zooming, the extent of the object captured clearly decreases as every sensor is limited by its FOV which dictates a stand-off distance as described in Figure 13. This is well simulated in Figure 7(b)(right).

5.2.3. Experimental results. To validate the hypothesis from our simulations, we ran the same experiments on a Turtlebot with our sensor mounted on it and scanned the cutout of a cross and circle separately in three different modes at 5 different distance ranges from 22 to 36 cm. We choose symmetric objects so that the angular resolution becomes equal, simplifying our analysis. Figure 7(c) and (d) show the scans and the graphs for the cross and the circle objects, respectively. The top row of Figure 7(c) shows the reconstructions achieved by mode A (left) fixed resolution at furthest distance of 36 cm, mode B (center) zooming from 36 cm distance, and again mode A (right) fixed resolution at the nearest distance of 22 cm. We can clearly see the superiority of mode B over mode A as it is able to recover a comparable spatial density from a stand-off distance of 36 cm as it would have in mode A but from a closer distance of 24 cm. This is also clearly evident in the spatial density graph in bottom row center left. In the bottom row, we measure angular resolution (left), spatial density (center left), scanning time (center right), and scanning efficiency (right).

Even though the angular resolutions of modes A and C remain constant over distance, mode C still achieves a better angular resolution. Mode B demonstrates the ability to change angular based on stand-off distance. We observe that for mode A, the scanning efficiency falls off with distance, but for modes B and C, it remains reasonably constant. Spatial density decreases for both mode A and mode C because of fixed angular resolution, but is higher for mode C showing its superiority. Modes A and B take the

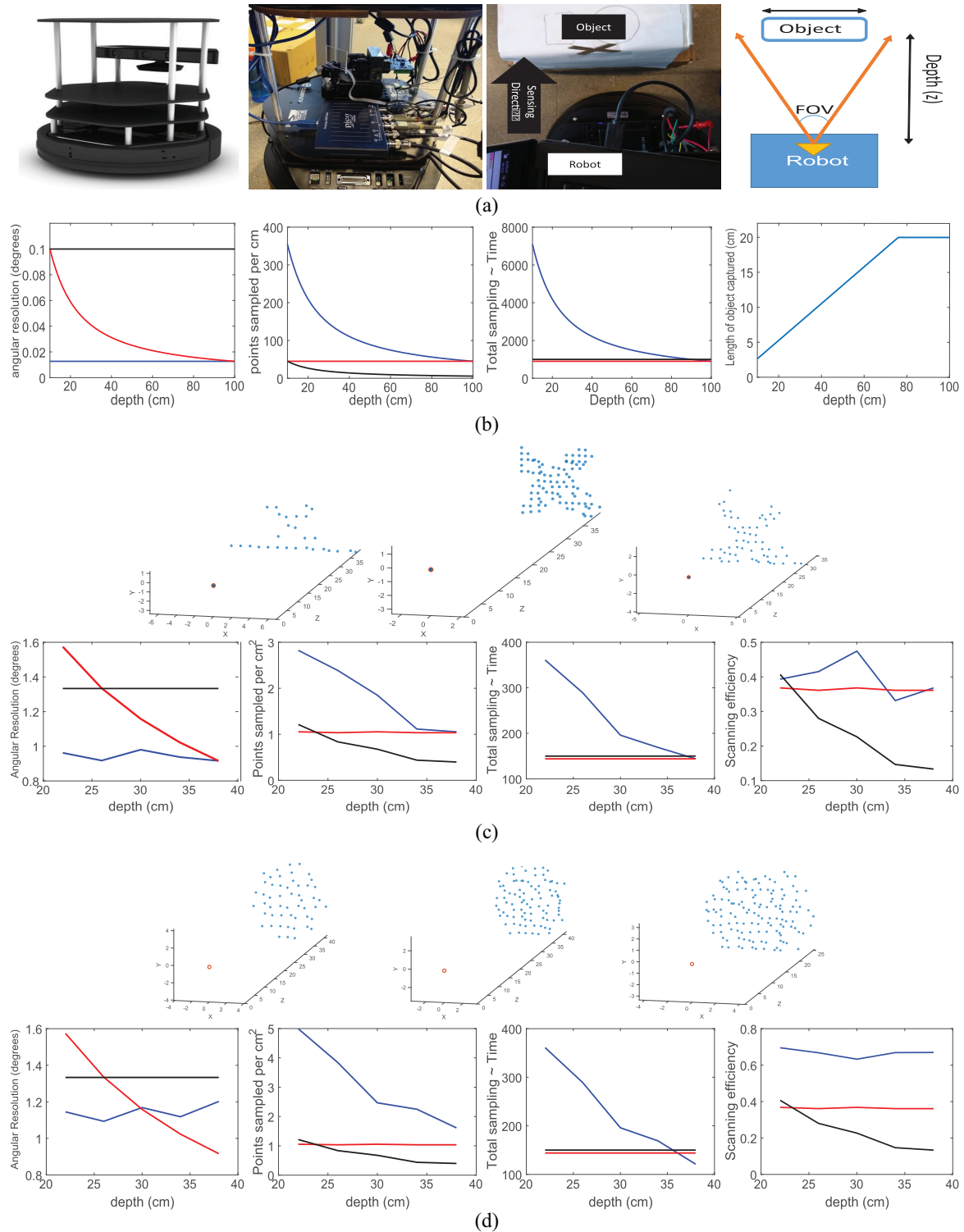


Fig. 7. Zooming for mobile robotic applications. We show the clear advantages that zooming provides over traditional fixed angular resolution sensors in terms of spatial density, angular resolution, scanning efficiency, and robustness at different scan distances. (a) Our experimental setup. In (b), we simulate the effects of different modes. In (c) and (d), we demonstrate the clear advantage of modes B and C over traditional sensors simulated in mode A. Modes A, B, and C are shown as black, red, and blue lines, respectively. In both (c) and (d), the top row shows the reconstructions achieved by mode A (left) fixed resolution at furthest distance of 36 cm, mode B (center) zooming from 36 cm distance, and again mode A (right) fixed resolution at the nearest distance of 22 cm. We conclude qualitatively that mode B (zooming) allows us to perform 3D imaging from a far stand-off distance with the same spatial density as the robot would have achieved in mode A (fixed resolution) from a closer distance. From the bottom rows of (c) and (d), we quantitatively conclude the advantage of modes B and C over mode A in terms of angular resolution (left), spatial density (center left), scanning time (center right), and scanning efficiency (right) at different depths.

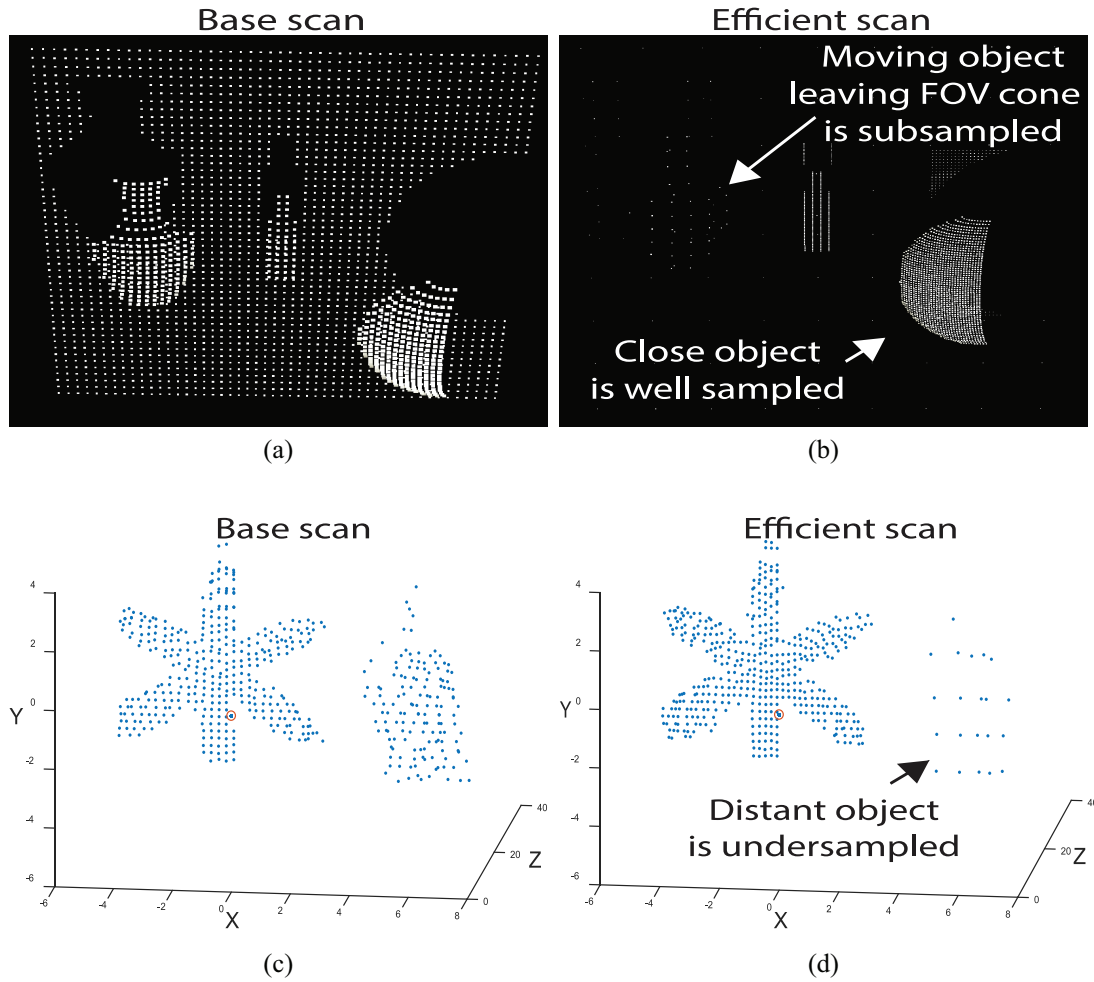


Fig. 8. Energy-aware adaptive sampling: (a) energy-efficient LiDAR simulation; (b) energy-efficient LiDAR with real scenes. We augment our directional control algorithm for adaptive TOF with physical constraints that capture the energy budget of the system. Here we use two constraints: the inverse fall-off of light beam intensity and the motion of scene objects with respect to the sensor FOV. In (a), we show simulations of three objects, one of which is given a motion perpendicular to the optical axis (i.e., leaving the sensor FOV cone). Compared with the base scan (left), the efficiency scan reduces sampling on objects that move beyond the FOV cone and distant objects, despite their complexity. In (b), we show a real example using our MEMS mirror-based sensor, where, again, distant objects are subsampled.

same amount of scanning time, but for mode C it could be as much as three times as high for objects in close range. Finally, scanning efficiency for modes B and C is almost always higher than mode A. In conclusion, both mode B and mode C have clear advantages in spatial density and scanning efficiency over mode A, which was a simulation of commercially available scanning TOF sensors using our sensor. Further, modes B and C represent fairly simple zooming rules, and these already demonstrate improved perception. We envision complex zooming in policies based on application and the ROI, allowing for dramatic improvements to previously proposed algorithms for robot perception. Although, we do not demonstrate usage of past information we can envision such online implementations demonstrating temporal intelligence for the case of active SLAM which is touched upon in Section 7.

5.3. Energy-aware adaptive sampling

In its current form, the adaptive TOF sensing algorithm only uses a geometric goodness metric. To augment the algorithm for mobile-based platforms, we wish to include multiple, say n , physical constraints into the metric. Therefore, we redefine the metric as $M(B_i) \in \mathbf{R}^n$, where B_i is the i th box in the tessellated current scan.

To illustrate the redefined metric, we point out differences between adaptive sensing when compared with adaptive sampling literature in image processing and graphics. First, for a given pulse signal and desired SNR, distant objects require more pulses. Therefore, geometric complexity must trade-off with range, and a distant, intricate object may not be sampled at the required resolution, to save energy. Second, temporal relevance matters, and a nearby, intricate object that is rapidly moving out of the FOV need

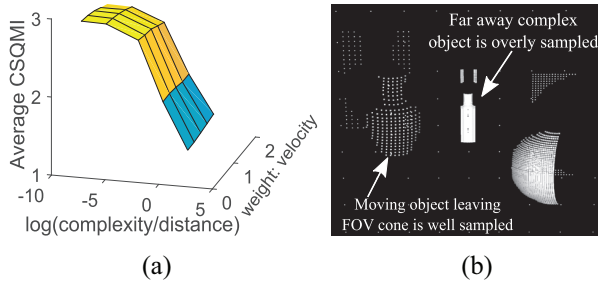


Fig. 9. Design parameters for energy-aware adaptive sampling: (a) variation in energy aware parameters; (b) scan with the worst CSQMI. We heuristically determine the parameters that decide the relative weight between the complexity and distance from the sensor as well as the importance given to clusters moving out of the FOV. In (a), we vary these parameters and observe the trend in average CSQMI over probable scans for the base scan in Figure 8(a). In (b), we scan with the worst CSQMI parameters, showing more samples on the distant complex object.

not be sampled at high resolution. Third, unlike virtual scenes, free space must be sampled periodically, because new obstacles may emerge. Finally, the sensor's measurement rate implies finite samples that must be shared across all objects, complex or simple.

The issues of free space and infinite samples are already handled by the adaptive algorithm described in Section 4.6, and we augment it with two new metrics in $M(B_i)$. The first takes into account the dissipation of the laser, and scores distant objects by two-way inverse square reduction in radiance, or $\frac{1}{Z^2}$. The second is simply a scaled version of the object's velocity $\lambda \vec{v}$, where λ is 1 if the direction \vec{v} is contained in the FOV cone of the sensor, and zero otherwise.

Figure 8 shows both simulated and real scenes, where objects are at different distances from the sensor. In the simulated scene, the vase is given a motion away from the sensor's visual cone. In the first column, we see the base scan of the scene, where samples are taken equiangularly. Applying the physical restrictions discussed above and using the adaptive algorithm described in Section 4.6 produces the results in the second column, where samples are reduced to save energy consumption and time.

5.3.1. Design parameters for energy-aware adaptive sampling. The various parameters of the algorithm described previously are the weights in the optimization, i.e., the penalty for motion out of the FOV and the trade-off between geometric goodness (complexity) and distance from the sensor origin. We analyze the effect of varying these parameters by observing the variation in CSQMI gain over probable scans for the simulation example described in Figure 8(a). In Figure 9(a), we observe that the CSQMI gain for probable scans is high for parameters where complexity is weighed more against the distance metric. The CSQMI gain value does not vary much on varying the penalty for clusters perpendicularly with respect to the FOV. In

Figure 9(b), we scan the scene using the trajectories generated for the worst CSQMI gain values, and we observe a distant object (bottle) is scanned thoroughly (wasting energy) owing to its complexity.

5.4. Edge sensing for gradient-based reconstruction

Gradient-based methods (Pérez et al., 2003) have had significant impact on vision, graphics, and imaging. Given a base scan of the scene, we can focus our sensor to place samples only on regions of high-frequency changes in depth. Placing all our samples in these regions, over the same time it took to previously scan the entire scene, produces more robust data because averaging can reduce noise in these edge regions.

Our goal is to estimate scene depths Z , from a small number of captured depths \hat{Z} . A popular solution is to minimize some norm between the numerically computed real and estimated x and y gradients. Formulating this for our scenario,

$$\min_Z \left\| \frac{d\hat{Z}}{dx} - \frac{dZ}{dx} \right\|^2 + \left\| \frac{d\hat{Z}}{dy} - \frac{dZ}{dy} \right\|^2 \quad (5)$$

Note that the minimization estimates scene depth Z , which has values outside the sparse locations where we have measurements, i.e., it is a full scene reconstruction. In Figure 10 we show a real example of gradient-based reconstruction for a scene with two planes at 27 and 37 cm. We captured a base scan in Figure 10(a) of the scene and, using its depth gradients, captured a new set of measurements along edges (Figure 10(b)). These were used with a widely available gradient reconstruction method (Harker and O'Leary, 2015), which reduced the corresponding RSMEs in Figure 10(e) by a third.

5.4.1. Discussion of advantages and brief qualitative comparisons to other reconstruction methods. Our approach offers a choice of trade-offs compared with standard, equiangular LiDAR reconstruction. For example, in the same time period, one could focus the samples onto the depth edges and recover the scene quickly. On the other hand, if time is not an issue, fewer samples are needed to achieve the same reconstruction quality, because these can be averaged only on the edges (rather than throughout the scene).

Finally, the approach and sensor design we have proposed focuses all the light into one LiDAR "dot" and scans it onto ROIs. Other active reconstruction approaches, such as phase-based TOF sensors, structured light sensors, or photometric stereo approaches, require illuminating all the scene's points in parallel. This allows for faster capture compared with our scanning technique, yet, for a fixed power budget, the illumination is spread over the entire scene FOV, in contrast to our focused LiDAR "dot." Our advantage in SNR, rather than

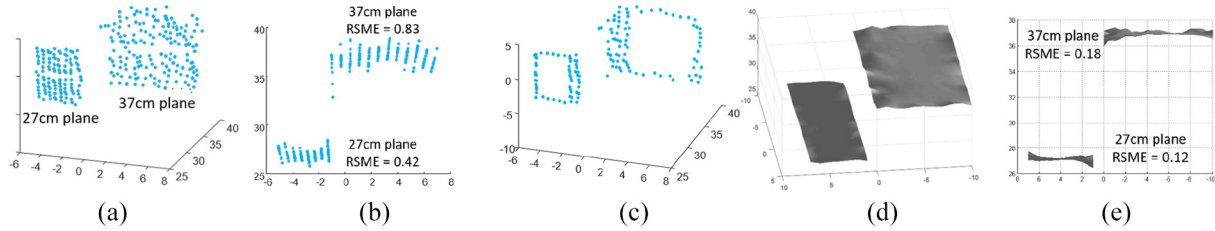


Fig. 10. Gradient-based reconstruction. Directional control allows a capture of the scene, where samples are only made in high-frequency (i.e., edge) regions of the scene. In (a), we see the original scan of the two planes, and (b) illustrates their noise levels. In (c), we directly capture only edge regions, placing the same number of samples as in (a) in high-frequency areas, improving averaging and reducing error. We use these edges with a gradient-based reconstruction algorithm to recover the meshes in (d). Note that the noise levels are reduced significantly, as shown in (e).

time, is particularly relevant for MEMS modulated LiDAR. First, MEMS motion can be very fast, reducing the disadvantage of time in scanning sensors. Second, the aperture of the system is the small MEMS mirror area, and therefore concentrating light into a dot improves SNR and range. In conclusion, our approach offers a different, scanning-based strategy to scene reconstruction, and the use of MEMS mirrors, with fast speed and foveated sensing, reduces the disadvantages of scanning.

6. Gazebo simulation

As the sensor proposed in this work is relatively new, obtaining a physical sensor for experiments can be challenging. An alternate way to enable quick use of such sensors is simulation. However, most sensor implementations in simulation have been passive. For active perception, the sensor is usually added to a moveable platform, which is then controlled through user input. We could not find any simulations for an adaptive sensor such as that described in this work. We therefore created a simulation for the sensor to enable others to perform experiments with this new kind of sensor.

We used the Gazebo simulation platform (Koenig and Howard, 2004) because it:

- supports various physics engines, ODE, Bullet, DART, and SimBody;
- integrates well with ROS;
- is widely used by the robotics community, both academia and industry;
- is open source and is widely available.

The simulation enables users to dynamically control different aspects of the sensor including the horizontal and vertical resolution, minimum and maximum angles, as well as the minimum and maximum range and noise parameters. Currently, the simulation supports Gaussian noise. However, it can be extended to different noise models. Figure 11 shows the sensor in the simulated environment. The image on the left shows the sensor initialized with the

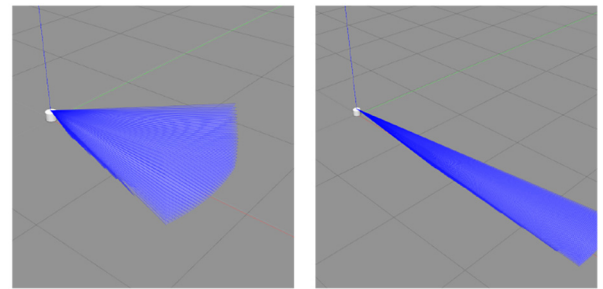


Fig. 11. Adaptive LiDAR sensor simulated in Gazebo. Sensor initialized with default parameters (left). Sensor parameters are changed through user input (right).

Table 1. Programming API to control various parameters on the sensor

Method	Parameter affected
set_hminangle	Horizontal minimum angle
set_hmaxangle	Horizontal maximum angle
set_hsamples	Horizontal number of samples
set_hres	Horizontal resolution
set_vminangle	Vertical minimum angle
set_vmaxangle	Vertical maximum angle
set_vsamples	Vertical number of samples
set_vres	Vertical resolution
set_rmin	Minimum range
set_rmax	Maximum range
set_rres	Range resolution
set_nmean	Noise mean
set_nstd	Noise standard deviation

default parameters. The image on the right shows the result of a user input changing the parameters of the sensor. An example program that alters the parameters of the adaptive LiDAR sensor during execution is shown in Listing 1. The available API methods and the parameters affected by them are listed in Table 1.

The code and assets for the simulation can be found at: <https://github.com/droneslab/AdaptiveLidarGazebo>
Simulation

Listing 1 An example program altering parameters of sensor

With an active LiDAR, such as that proposed in this work, mobile robots will have the ability to reduce the entropy in the map without the need for movement by vir-

```
#include "gazebo/gazebo.hh"
#include "gazebo/common/common.hh"
#include "gazebo/transport/transport.hh"
#include "gazebo/physics/physics.hh"
#include "gazebo_msgs/msgs.hh"
#include "adaptive_lidar_request.pb.h"
int main(){
  AdaptiveLidarRequest request;
  // Horizontal
  request.set_hminangle(-0.52);
  request.set_hmaxangle(0.52);
  request.set_hsamples(32);
  request.set_hres(1.0);
  // Vertical
  request.set_vminangle(-0.52);
  request.set_vmaxangle(0.52);
  request.set_vsamples(32);
  request.set_vres(1.0);
  // Range parameters
  request.set_rmin(0.4);
  request.set_rmax(2.4);
  request.set_rres(0.1);
  // Noise parameters
  request.set_nmean(0.0);
  request.set_nstd(0.1);
  gazebo::transport::NodePtr node(
    new gazebo::transport::Node());
  node->Init("default");
  gazebo::transport::PublisherPtr pub =
    node->Advertise<AdaptiveLidarRequest>(
      "~/collision_map/command");
  pub->WaitForConnection();
  pub->Publish(request);
  gazebo::transport::fini();
  return 0;
}
```

tue of its capability to zoom in on portions of the map. The control inputs to reduce map entropy in local regions, by zooming in, can be made independent of control inputs for navigation. Therefore, the objective of optimizing for mapping and localization accuracy can be decoupled from other objectives such as minimizing distance traveled.

7. Discussion

Although we show only static scene reconstructions, our adaptive angular framework impacts any scanning TOF sensor.

7.1. Deblurring trade-off

Given a minimum, required incident radiance at the photo-detector, our sensor range Z and FOV Θ are inversely

proportional, $Z^2 \propto \frac{1}{\tan(\frac{\Theta}{2})}$ (Figure 13(a)). Our results have significant scope for improvement in measurement SNR of the reconstructions, and we will focus on better optical designs in the future.

7.2. System performance

In Figure 13(b), we compare the ability to induce desired sample density on targets. For conventional sensors, as the density increases, the robot-target distance goes to zero. For our sensor design, a stand-off distance is possible because we can concentrate samples on the target.

7.3. Efficiency/power reduction applications

We will use energy-efficient adaptive sensing for unmanned aerial vehicles (UAVs) and other power-constrained robots to place the samples on nearby obstacles and targets, accruing power savings.

7.4. Toward a foveated sensor model

A probabilistic sensor model for our scanning LiDAR is crucial for a variety of intelligent robot applications, such as SLAM (Thrun et al., 2005). Here we show the first step toward creating that model for our new sensor design.

Most TOF sensors provide readings that follow a normal distribution. We now discuss this normal distribution for the five-plane calibration of our sensor described earlier. Our model consists of fitting Gaussian parameters ($\mu(\theta, \phi, Z)$ and $\sigma(\theta, \phi, Z)$) that vary with direction (θ, ϕ) and depth Z . Gaussian is a simple and popular choice for beam models for range sensors. Our experiments show that this is a fair choice. Figure 12(a)(left) shows results from multiple depth measurements made to a plane placed 25 cm from the sensor. We observe that the measurements can be fit with a Gaussian. From Figure 12(a)(center left), we observe that the standard deviation changes as the plane is moved farther away from the sensor. Therefore, the sensor model must account for the change in depth. We also observe that the measurements depend on the angle of the beam. Figure 12(a)(center right) shows measurements from 36 viewing directions and we observe that a Gaussian model fits well with these measurements. For a particular fit of the model in Figure 12(a)(center right), we demonstrate the reconstruction (in simulation) of a 3D “tree” object as shown in Figure 12(b). Figures 12(a)(right) and 12(b) show the decrease in error as we take multiple scans and average. Note that there is still some bias between the model and real sensor, which will need further calibration. However, the trend of these curves show that our model does correctly follow the sensor behavior.

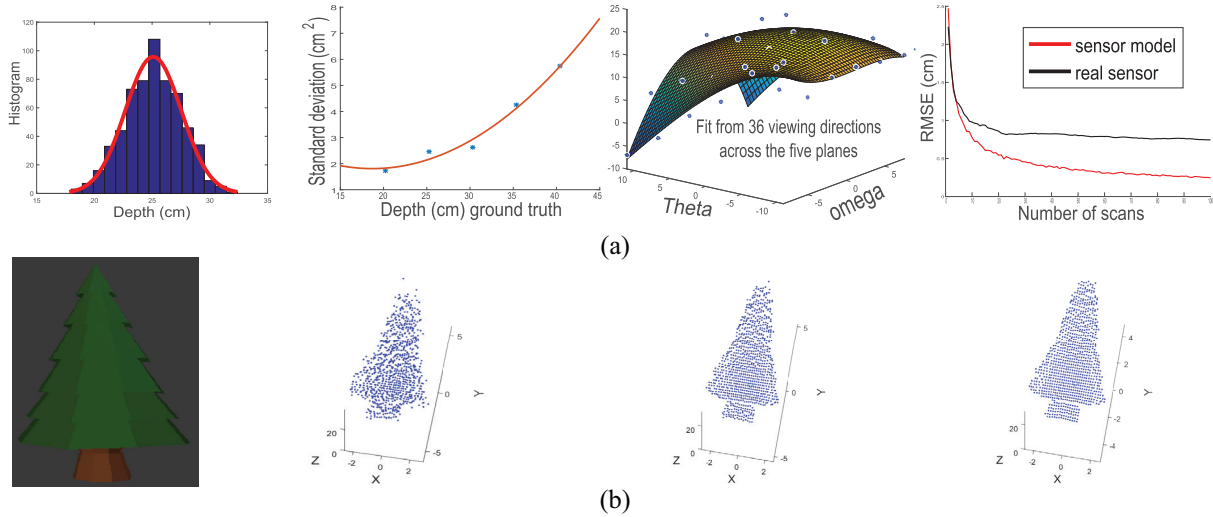


Fig. 12. Toward a foveated sensor model. In (a), we learn the sensor model for our TOF sensor, and in (b), we simulate scans using the learned sensor model to show the improvement of reconstructions through simple averaging.

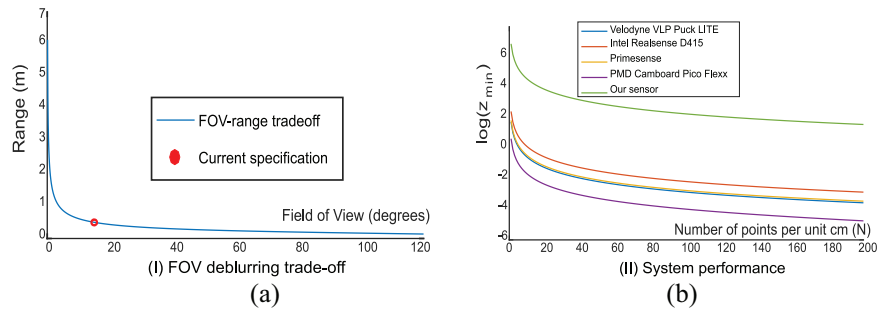


Fig. 13. In (a), we depict the deblurring trade-off in our setup, where increasing FOV results in reduced SNR and range. In (b), we compare our LiDAR zoom with other available depth sensors that have fixed acuity.

7.5. Performance improvements in mobile robots

Localization, mapping, and navigation accuracy is improved with increased sensor resolution. While improved mapping accuracy is a direct result of increased resolution, improvements in localization and navigation accuracy are a result of better feature matching, scan matching, and loop closures. In this context, active SLAM is the problem of finding trajectories that reduces mapping and localization error through exploration. Typically, implementations of active SLAM use an information gain formulation to reduce map entropy. Some implementations such as Zhang et al. (2006) use formulations that balance the objective of reducing the entropy in the map while minimizing the cost of navigation. Other implementations such as Mu et al. (2016) plan control inputs that increase the number of features in the map. We note that in real-world scenarios, trajectories that conform to other constraints such as minimizing distance, minimizing battery usage, etc., are usually desired or necessary. Incorporating such constraints into active SLAM would lead to complex joint optimization problems.

With an active LiDAR, such as that proposed in this work, mobile robots will have the ability to reduce the entropy in the map without the need for movement by virtue of its capability to zoom in on portions of the map. The control inputs to reduce map entropy in local regions, by zooming in, can be made independent of control inputs for navigation. Therefore, the objective of optimizing for mapping and localization accuracy can be decoupled from other objectives such as minimizing distance travelled.

8. Conclusion

In this work, we have presented the design of a scanning depth sensor, which can control its angular resolution over its FOV. Such a sensor is different from existing depth sensors in vision/robotics that typically have a fixed angular resolution, and opens up novel possibilities in design of algorithms in localization, mapping, exploration, and manipulation in robotics.

We continue to work on improvements to the hardware and software implementations to allow real-time capture

and processing of dynamic scenes with robot motion. If this is enabled, then there is no barrier to demonstrating joint robot pose and LiDAR zoom that allow for efficient SLAM. Finding the efficiencies in capture time and robot motion for such algorithms, when compared with conventional SLAM, is one of the potential directions of future work.


Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The University of Florida authors have been partially supported by the National Science Foundation through NSF IIS: 1909729, and the Office of Naval Research through ONR N00014-18-1-2663. C. Adhivarahan was partially supported through the Community of Excellence on Sustainable Manufacturing and Advanced Robotics Technologies (SMART CoE) at University at Buffalo and NSF #1514395. Karthik Dantu was partially supported by NSF #1514395 and NSF #1846320.

Notes

- 1 Available at: http://focus.ece.ufl.edu/wp-content/uploads/Adaptiv_Lidar_T-Ro.mp4
- 2 See <https://www.clearpathrobotics.com/turtlebot-2-open-source-robot/>

ORCID iD

Zaid Tasneem  <https://orcid.org/0000-0001-6515-4172>

References

- Achar S, Bartels JR, Whittaker WL, Kutulakos KN and Narasimhan SG (2017) Epipolar time-of-flight imaging. *ACM Transactions on Graphics* 36(4): 37.
- Badki A, Gallo O, Kautz J and Sen P (2017) Computational zoom: A framework for post-capture image composition. *ACM Transactions on Graphics* 36(4): 46.
- Bailey T and Durrant-Whyte H (2006) Simultaneous localization and mapping (SLAM): Part II. *IEEE Robotics and Automation Magazine* 13(3): 108–117.
- Bajcsy R (1988) Active perception. *Proceedings of the IEEE* 76(8): 996–1005.
- Bandopadhyay A, Aloimonos J and Weiss I (1988) Active vision. *International Computer Vision* 1(4): 333–356.
- Beckers JM (1993) Adaptive optics for astronomy: Principles, performance, and applications. *Annual Review of Astronomy and Astrophysics* 31(1): 13–62.
- Beraldin JA, Blais F, Courmoyer L, Godin G, Rioux M and Taylor J (2003) Active 3D sensing. Techcanl Report, Scuola Normale Superiore Pisa. Available at: <http://www1.cs.columbia.edu/allen/PHOTOPAPERS/beraldin.pdf>
- Campbell AT III and Fussell DS (1990) Adaptive mesh generation for global diffuse illumination. *ACM SIGGRAPH Computer Graphics* 24: 155–164.
- Carlone L and Karaman S (2017) Attention and anticipation in fast visual-inertial navigation. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 3886–3893.
- Charrow B, Kahn G, Patil S, et al. (2015) Information-theoretic planning with trajectory optimization for dense 3d mapping. In: *Robotics: Science and Systems*.
- Del Bimbo A, Dini F, Lisanti G and Pernici F (2010) Exploiting distinctive visual landmark maps in pan-tilt-zoom camera networks. *Computer Vision and Image Understanding* 114(6): 611–623.
- Denzler J and Brown CM (2002) Information theoretic sensor data selection for active object recognition and state estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(2): 145–157.
- Deselaers T, Dreuw P and Ney H (2008) Pan, zoom, scan-time-coherent, trained automatic video cropping. In: *IEEE Conference on Computer Vision and Pattern Recognition*, 2008 (CVPR 2008). IEEE, pp. 1–8.
- Durrant-Whyte H and Bailey T (2006) Simultaneous localization and mapping: part I. *IEEE Robotics and Automation Magazine* 13(2): 99–110.
- Fergus R, Torralba A and Freeman WT (2006) Random lens imaging. *MIT CSAIL Technical Report* 2006-058.
- Flatley TP (2015) Spacecube: A family of reconfigurable hybrid on-board science data processors. In: *NASA/ESA Conference on Adaptive Hardware and Systems (AHS-2012)*.
- Goodman JW (2005) *Introduction to Fourier Optics*. Roberts and Company Publishers.
- Gschwandtner M, Kwitt R, Uhl A and Pree W (2011) BlenSor: Blender sensor simulation toolbox. In: *International Symposium on Visual Computing*. Berlin: Springer, pp. 199–208.
- Gupta M, Nayar SK, Hullin MB and Martin J (2015) Phasor imaging: A generalization of correlation-based time-of-flight imaging. *ACM Transactions on Graphics* 34(5): 156.
- Halterman R and Bruch M (2010) Velodyne HDL-64e lidar for unmanned surface vehicle obstacle detection. Technical Report, Space and Naval Warfare Systems Center, San Diego, CA.
- Harker M and O’Leary P (2015) Regularized reconstruction of a surface from its measured gradient field. *Journal of Mathematical Imaging and Vision* 51(1): 46–70.
- Heide F, Hullin MB, Gregson J and Heidrich W (2013) Low-budget transient imaging using photonic mixer devices. *ACM Transactions on Graphics* 32(4): 45.
- Hollinger GA and Sukhatme GS (2014) Sampling-based robotic information gathering algorithms. *The International Journal of Robotics Research* 33(9): 1271–1287.
- Izadi S, Kim D, Hilliges O, et al. (2011) 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*. New York: ACM Press, pp. 559–568.
- Jones A, McDowall I, Yamada H, Bolas M and Debevec P (2007) Rendering for an interactive 360 degree light field display. In: *SIGGRAPH*. New York: ACM Press.
- Julian BJ, Karaman S and Rus D (2014) On mutual information-based control of range sensing robots for mapping applications. *The International Journal of Robotics Research* 33(10): 1375–1392.
- Kasturi A, Milanovic V, Atwood BH and Yang J (2016) UAV-borne LiDAR with MEMS mirror-based scanning capability. *Proceedings of SPIE* 9832: 98320M.
- Koenig N and Howard A (2004) Design and use paradigms for Gazebo, an open-source multi-robot simulator. In: *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vol. 3, pp. 2149–2154

- Koppal SJ, Gkioulekas I and Young T (2013) Toward wide-angle microvision sensors. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12: 2982–2996.
- Koppal SJ, Yamazaki S and Narasimhan SG (2012) Exploiting DLP illumination dithering for reconstruction and photography of high-speed scenes. *International Journal of Computer Vision* 96(1): 125–144.
- Krastev KT, Van Lierop HW, Soemers HM, Sanders RHM and Nellissen AJM (2013) Mems scanning micromirror. US Patent 8,526,089.
- Levin A, Fergus R, Durand F and Freeman WT (2007) Image and depth from a conventional camera with a coded aperture. *ACM Transactions on Graphics* 26: 70.
- Milanović V, Kasturi A, Siu N, Radojičić M and Su Y (2011) “MEMSeye” for optical 3D tracking and imaging applications. In: *2011 16th International Solid-State Sensors, Actuators and Microsystems Conference (TRANSDUCERS)*. IEEE, pp. 1895–1898.
- Milanović V, Kasturi A, Yang J and Hu F (2017) A fast single-pixel laser imager for VR/AR headset tracking. *Proceedings of SPIE* 10116: 101160E–1.
- Mu B, Giamou M, Paull L, Agha-mohammadi A, Leonard J and How J (2016) Information-based active SLAM via topological feature graphs. In: *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 5583–5590.
- Murray D and Jennings C (1997) Stereo vision based mapping and navigation for mobile robots. In: *Proceedings of International Conference on Robotics and Automation*, Vol. 2. IEEE, pp. 1694–1699.
- Nayar SK, Branzoi V and Boulton TE (2006) Programmable imaging: Towards a flexible camera. *International Journal of Computer Vision* 70(1): 7–22.
- Nehab D, Sander PV and Isidoro JR (2006) The real-time reprojection cache. In: *ACM SIGGRAPH 2006 Sketches*. New York: ACM Press, p. 185.
- Ng R (2005) Fourier slice photography. *ACM Transactions on Graphics* 24: 735–744.
- O’Toole M, Achar S, Narasimhan SG and Kutulakos KN (2015) Homogeneous codes for energy-efficient illumination and imaging. *ACM Transactions on Graphics* 34(4): 35.
- O’Toole M, Heide F, Xiao L, Hullin MB, Heidrich W and Kutulakos KN (2014) Temporal frequency probing for 5D transient analysis of global light transport. *ACM Transactions on Graphics* 33(4): 87.
- Paletta L, Prantl M and Pinz A (2000) Learning temporal context in active object recognition using Bayesian analysis. In: *Proceedings 15th International Conference on Pattern Recognition (ICPR-2000)*, Vol. 1. IEEE, pp. 695–699.
- Pérez P, Gangnet M and Blake A (2003) Poisson image editing. *ACM Transactions on Graphics* 22(3): 313–318.
- Petersen KE (1980) Silicon torsional scanning mirror. *IBM Journal of Research and Development* 24(5): 631–637.
- Poulton CV, Yaacobi A, Cole DB, et al. (2017) Coherent solid-state LiDAR with silicon photonic optical phased arrays. *Optics Letters* 42(20): 4091–4094.
- Principe JC, Euliano NR and Lefebvre WC (2000) *Neural and Adaptive Systems: Fundamentals Through Simulations*, Vol. 672. New York: John Wiley & Sons, Inc.
- Qi CR, Su H, Mo K and Guibas LJ (2017) PointNet: Deep learning on point sets for 3D classification and segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 652–660.
- Raskar R, Agrawal A and Tumblin J (2006) Coded exposure photography: Motion deblurring using fluttered shutter. *ACM Transactions on Graphics* 25(3): 795–804.
- Raskar R, Welch G, Cutts M, Lake A, Stesin L and Fuchs H (1998) The office of the future: A unified approach to image-based modeling and spatially immersive displays. In: *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*. New York: ACM Press, pp. 179–188.
- Rosman G, Rus D and Fisher JW (2016) Information-driven adaptive structured-light scanners. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 874–883.
- Sandner T, Baulig C, Grasshoff T, et al. (2015) Hybrid assembled micro scanner array with large aperture and their system integration for a 3D TOF laser camera. In: *MOEMS and Miniaturized Systems XIV (Proc. SPIE, Vol. 9375)*. International Society for Optics and Photonics, p. 937505.
- Scharstein D and Szeliski R (2003) High-accuracy stereo depth maps using structured light. In: *Proceedings 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 1. IEEE.
- Stann BL, Dammann JF, Del Giorno M, et al. (2014) Integration and demonstration of MEMS-scanned LiDAR for robotic navigation. *Proceedings of SPIE* 9084: 90840J.
- Tasneem Z, Wang D, Xie H and Koppal SJ (2018) Directionally controlled time-of-flight ranging for mobile sensing platforms. In: *Robotics: Science and Systems XIV*.
- Terzopoulos D and Vasilescu M (1991) Sampling and reconstruction with adaptive meshes. In: *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1991 (CVPR’91)*. IEEE, pp. 70–75.
- Thrun S, Burgard W and Fox D (2005) *Probabilistic Robotics*. Cambridge, MA: MIT Press.
- Tyson RK (2015) *Principles of Adaptive Optics*. Boca Raton, FL: CRC Press.
- Velten A, Wu D, Masia B, et al. (2016) Imaging the propagation of light through scenes at picosecond resolution. *Communications of the ACM* 59(9): 79–86.
- Will PM and Pennington KS (1972) Grid coding: A novel technique for image processing. *Proceedings of the IEEE* 60(6): 669–680.
- Yang B, Zhou L, Zhang X, Koppal S and Xie H (2017) A compact MEMS-based wide-angle optical scanner. In: *2017 International Conference on Optical MEMS and Nanophotonics (OMN)*. IEEE, pp. 1–2.
- Yang H, Xi L, Samuelson S, Xie H, Yang L and Jiang H (2013) Handheld miniature probe integrating diffuse optical tomography with photoacoustic imaging through a MEMS scanning mirror. *Biomedical Optics Express* 4(3): 427–432.
- Zabatani A, Surazhsky V, Sperling E, et al. (2019) Intel RealSense SR300 coded light depth camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. DOI: 10.1109/TPAMI.2019.2915841.
- Zhang N, Li M and Hong B (2006) Active mobile robot simultaneous localization and mapping. In: *2006 IEEE International Conference on Robotics and Biomimetics*, pp. 1676–1681.
- Zhang Y, Xiong Z, Cong P and Wu F (2014) Robust depth sensing with adaptive structured light illumination. *Journal of Visual Communication and Image Representation* 25(4): 649–658.