2019

Computational Imaging Systems for High-speed, Adaptive Sensing Applications

Yangyang Sun
University of Central Florida

COMPUTATIONAL IMAGING SYSTEMS FOR HIGH-SPEED, ADAPTIVE SENSING APPLICATIONS

by

YANGYANG SUN
B.S. Nanjing University, 2013
M.S. University of Central Florida, 2016

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the College of CREOL/The College of Optics & Photonics
at the University of Central Florida
Orlando, Florida

Fall Term
2019

Major Professor: Shuo Pang

ABSTRACT

Driven by the advances in signal processing and ubiquitous availability of high-speed low-cost computing resources over the past decade, computational imaging has seen growing interest. Improvements on spatial, temporal, and spectral resolutions have been made with novel designs of imaging systems and optimization methods. However, there are two limitations in computational imaging. 1), Computational imaging requires full knowledge and representation of the imaging system called forward model to reconstruct the object of interest faithfully. This limits the applications in the systems with parameterized unknown forward model such as range imaging systems. 2), The regularization in the optimization process incorporates strong assumptions which may not accurately reflect the *a priori* distribution of the object. To overcome these limitations, we propose 1) novel optimization frameworks for applying computational imaging on active and passive range imaging systems and achieve 5-10 folds improvement on temporal resolution in various range imaging systems; 2) a data-driven method for estimating the distribution of high dimensional objects and a framework of adaptive sensing for maximum information gain. The adaptive strategy with our proposed method outperforms Gaussian process-based method consistently. The work would potentially benefit the high-speed 3D imaging applications such as autonomous driving and adaptive sensing applications such as low-dose adaptive computed tomography(CT).

Keywords: Range Imaging, Computational Imaging, Adaptive Sensing, Sensor Placement

ACKNOWLEDGMENTS

Throughout my career as a graduate student, I have received a great deal of supports. I would like to thank my supervisor Dr. Shuo Pang for his invaluable expertise in formulating the research topics and the methodology.

I gratefully acknowledge my supervisor from my intern at Nokia Bell Labs, Dr. Xin Yuan. Thank you for your excellent cooperation and for all the opportunities I was given to conduct my research at Bell Labs.

I would like to thank my wife for lending me her sympathetic ear and my daughter for those happy distractions to rest my mind outside the research.

TABLE OF CONTENTS

LIST OF FIGURES vii

CHAPTER 1: INTRODUCTION 1

1.1 History of Image Sensors 1

1.2 Computational Imaging 1

CHAPTER 2: COMPUTATIONAL IMAGING FOR HIGH-SPEED RANGE IMAGING . 6

2.1 Range Imaging 6

2.2 Passive Computational Range Imaging 6

2.2.1 Forward Model 7

2.2.2 Reconstruction 8

2.2.3 Experimental Results 13

2.3 Active Computational Range Imaging 16

2.3.1 Forward Model 16

2.3.2 Reconstruction 17

2.3.3 Experimental Results 21

2.4 Limitation and Outlook 23

CHAPTER 3: COMPUTATIONAL IMAGING IN ADAPTIVE SENSING 26

3.1 Adaptive Sensing and Sensor Placement 26

3.1.1 Objective Function in Adaptive Sensing 28

3.1.2 Near Optimal Strategy:Two-step Uncertainty Network 30

3.1.3 Results and Discussion 32

3.2 Limitation and Outlook 37

CHAPTER 4: CONCLUSION 39
4.1 Summary of Contributions 39
4.2 Future Outlook 40

LIST OF REFERENCES 41

LIST OF FIGURES

Figure 2.1: System schematic (a) On the left-view optical path, Camera 1 records low-speed measurement IL. On the right-view optical path, high-speed right-view scene F_{RH} are encoded by the DMD with N distinct patterns M. The coded right-view scene is then relayed by lens Lrelay and recorded by Camera 2 within the exposure time to form the right-view measurement IR. (b) A photo of the setup 8

Figure 2.2: The flow chart of the reconstruction algorithm. The high-speed scene F_{RH} is reconstructed from the modulated measurement I_R using video compressive sensing inversion algorithm, TwIST. Then, the high-speed depth maps are estimated from I_L and F_{RH} by our one-to-N correspondence algorithm based on Graph Cut. 10

Figure 2.3: Reconstruction of a backward-moving ball. (a) Measurements from two optical paths in our system. The different traces of the motion blurs indicate the varying depth of the moving ball. (b) 1^{st}, 6^{th} and 10^{th} fames of reconstructed high-speed video from single measurement I_R. (c) 1^{st}, 6^{th} and 10^{th} fames of reconstructed high-speed depth map. (d) An overlay plot of 10 depth maps within a single exposure. The gradient of the color implies that the ball is moving away from our imaging system. 14

Figure 2.4: Reconstruction of an 800fps high-speed scene with two flying "shurikens". (a) Stereo imaging measurements. (b) Selected frames of reconstructed 800fps video. (c) 3D rendering of the high-speed scene. The rectangular shuriken rotated about 30° within the exposure time, and the triangular shuriken rotated about 20° 15

Figure 2.5: The forward model of our system. The projector projects high-speed masks onto the scene. The scene is modulated by $h^*(x,y,t)$, the variants of the original mask at different ranges. The camera integrates the modulated high-speed frames into one measurement. 17

Figure 2.6: Schematics of our system. The lateral offset of projector and camera is d. The distance between projector and camera along the z axis is l_0. θ denotes the projection angle. The focal lengths of the camera and the projector are not specified in the figure for concision. 19

Figure 2.7: Calibrated masks at different ranges. Same patterns of the 5^{th} mask projected at different ranges are highlighted with the red boxes. The shift of the red boxes indicates one modulation on the range. The scales of the zoom in patterns is the other modulation on the range. 20

Figure 2.8: The top row presents one frame of a 200 fps measurement and the reconstructed 1000 fps reflectance frames. The object is a disk rotating at 30 rounds per second. The red box indicates the position of the black tape in the first frame. The bottom row shows the ground truth of the 3D scene and the reconstructed depth map. A video of 1000-fps frames has been reconstructed from 200-fps measurement 24

Figure 3.1: (a) The diagram of sensor placement (black arrows) and TUN(red arrows). TUN consists of two steps: imagination step and inspection step to generate the possible measurements and evaluate the task-specific information gain at un-observed locations. (b) Graphical models of the two steps in TUN. Instances of measurement x_k at un-observed locations are generated in the imagination step. Then those generated measurements are used to evaluate the information gain for the task in inspection step. 27

Figure 3.2: Example of the generation step on 1D spectrum dataset. The dashed line is the true spectrum and the solid spots are the observations. The generated instances are in colorful solid lines. Left: 10 imagined spectrums based on single observation. Right: 10 imagined spectrums based on three observations. The variation in the generated samples is mainly from scales and is independent of the task. 28

Figure 3.3: Left: x-ray security screening system. 8 different directions(locations) uniformly distributed within 180° are available to illuminate the object . The object is randomly rotated along z axis with an obstacle. Additive Gaussian noise is adapted on the measurements. Right: examples of measurements at 8 locations. 32

Figure 3.4: The imagination step of TUN on high dimensional dataset. Left: The 3 instances of the generations at location 6 and 7 given the observations at location 4. The generated instances are different in labels(digit 0 and 7) and fonts, but they all keep the corner feature occurred in the observation.Middle: The generations given two observations at location 4 and 5. The generated instances are much more convergent as more information is extracted from the observations. Right: The ground truth of the measurements at location 6 and 7. 34

Figure 3.5: Task-specific feature distribution. We visualize the initial intermediate feature distribution in the inspector network using t-SNE. Clearly the features become more disperse at further locations from the observed location. The feature space is colored by locations. Region 1: Features of generated instances at the observed location. Region 2: Features at locations that are close to the observed location. Region 3: away from observed location. Region 3: Features at locations far from the observed locations. 35

Figure 3.6: Entropy estimation: (a) The remaining entropy at un-observed locations given a single observation at location 5. The optimal next location in this step is location 2 at which the expected remaining entropy is least. (b) As we obtain the measurement at location 2 following TUN, the estimated remaining entropy at all the un-observed locations are almost zeros, indicating that the model is quite confident about the label with those observations. 36

Figure 3.7: Accuracy and entropy at different numbers of inquiries for TUN, Gaussian process(GP) based and random sampling(RS) strategies. The advantages of sensor placement strategies starts at two observations. The proposed framework, TUN, outperforms GP and RS strategies consistently. 38

CHAPTER 1: INTRODUCTION

1.1 History of Image Sensors

Photography has a long history since 1839 introduced by Louis Daguerre[1]. The first generation of camera did not use film but light-sensitive chemicals that formed on the silver-plated copper sheet. Glass plate coated with photographic emulsion was introduced from 1850s as a less expensive alternative especially in astrophotography and electron micrography. The first plastic roll film was invented in 1889 which was made from highly flammable materials called nitrocellulose and became the standard theatrical 35 mm film[2]. The film cameras remained the domination of camera sensors until 2000s when the digital cameras supplanted them[3]. Modern digital cameras converted the two dimensional optical signal into digital signal at typically 30 frames per second on millions of pixels. The digital signal enables the post processing step in imaging such as denoising, white balancing, and contrast. Beyond the processing techniques on direct measured image, indirectly reconstructing the images from multiple measurements, termed computational imaging, such as computational microscopy[4, 5, 6], tomographic imaging, magnetic resonance imaging(MRI)[7], and coded aperture imaging[8] techniques have undergone significant improvement thanks to the advances of signal processing and the ubiquitous availability of high-speed computing resources.

1.2 Computational Imaging

Computational imaging models the imaging system as a function of the object f,

$$g = H(f) + n \tag{1.1}$$

where g is the observable measurement, n is the additive noise. The function H is called forward model which describes the physical process of the imaging system. Prior to any computations, the first thing is to calibrate the imaging system as well as the forward model H. It is inevitable that the calibrated forward model H will be deteriorated by the measurement errors and detection noises. However, it is still a better option than using the forward model built under ideal assumptions, especially when the system is complex. In this dissertation, we consider discretized measurement g and object f which are one dimensional vectors stretched from high dimensional data. The goal of computational imaging is to retrieve the object f from the measurement g inversely. This can be formulated as an optimization problem by defining and minimizing an objective function. Ideally, we would like to minimize the distance between the reconstruction and the true object $\mathscr{L}(\hat{f}, f)$. However, the true object f is not available. Thus, we minimize the distance between the synthetic measurement and the real measurement $\mathscr{L}(H(\hat{f}), g)$ where H is obtained from the system calibration. Given the stochastic nature of the noise n on the measurements, it is natural to use the negtive log-likelihood as the objective function. For instance, assume the noise is a Gaussian noise $n \sim \mathscr{N}(0, \sigma)$, the negtive log-likelihood of $\hat{f}$ can be evaluated by

$$\mathscr{L} = \frac{1}{2\sigma^2} \left\| H(\hat{f}) - g \right\|_2^2 + \frac{1}{2} \ln 2\pi\sigma^2 \tag{1.2}$$

Note that the second term is independent of $\hat{f}$. Thus, it will be discarded in the optimization. The coefficient in the first term before the L_2 norm is a constant. Through a maximum likelihood framework, the reconstruction f^* satisfies

$$f^* = \arg\min_{\hat{f}} \frac{1}{2} \left\| H(\hat{f}) - g \right\|_2^2 \tag{1.3}$$

Equation 1.3 only requires the forward model H and the real measurement g. If the forward model H is differentiable, this can be optimized through gradient based optimizations. The scalar

$\frac{1}{2}$ is for the convenience of taking derivative. It worth noting that different types of the noise distributions result in different forms of negative log-likelihoods. Equation 1.3 incorporates the intuition that the closer the synthetic measurement $H(\hat{f})$ to the real measurement g is, the better guess $\hat{f}$ is. However, this is not always true. Using Equation 1.3 as objective function showed poor performance in practice. To better regularize the reconstruction, different regularization terms are proposed such as total variance (TV)[9] and sparsity[10]. Adding the regularizations showed different degree of success in many applications[11, 12]. Formly, the reconstruction f^* optimizes

$$f^* = \arg\min_{\hat{f}} \frac{1}{2}\left\|H(\hat{f}) - g\right\|_2^2 + \lambda\Phi(\hat{f}) \tag{1.4}$$

where Φ is the regularizer on $\hat{f}$, and λ is the regularization parameter controlling the strength of the regularization. The first term in Equation 1.4 is the data term which enforces the reconstructed object $\hat{f}$ should generate measurements close to the observed g. The second term is the regularization term which enforces certain *a prori* structure of the object. Note that the regularization Φ is often selected to be convex for the convenience of the optimization.

There are many existing iterative methods to optimize Equation 1.4[13, 14, 15]. Most of them can be reformulated into proximal optimization methods[16]. It deals with the two terms,namely a differentiable data term and a convex regularization term, in Equation 1.4 separately. For example in proximal gradient method(PGM), the iteration can be written as

$$f^{k+1} = \boldsymbol{Prox}_{\lambda\Phi}(f^k - \frac{\gamma}{2}\nabla\left\|H(f^k) - g\right\|_2^2) \tag{1.5}$$

where $\boldsymbol{Prox}(\cdot)$ is the proximal operator, γ is the step size which can be determined through line search methods[17]. For $\Phi := L_1$ norm, the proximal operator is a soft thresholding operator. For $\Phi := L_2$ norm, the proximal operator is called a shrinkage operator. If the imaging system is a linear system, the forward model H can be represented by a matrix. Equation 1.5 can be written as

$$f^{k+1} = \boldsymbol{Prox}_{\lambda\Phi}(f^k - \gamma H^T(Hf^k - g)) \tag{1.6}$$

where H^T is the transpose matrix of H.

Most imaging systems such as spectral domain optical coherence tomography(SD-OCT), computed tomography(CT), Fourier ptychography, and compressive coded aperture imaging(CSSI) can be formulated into this framework with different forward models H. It is worth noting that Equation 1.6 assumes a linear imaging system in which the forward model can be represented by a matrix. In the case of non-linear imaging systems such as optical diffraction tomography(ODT) and phase retrieval imaging system, the forward model cannot be represented by a matrix but a differentiable function as shown in Equation 1.5.

Despite the success of computational imaging, there are two limitations. 1). The first limitation stems from the data term. The forward model H has to be known explicitly for taking the derivative in Equation 1.5. In practice, the forward model is calibrated from the real imaging setup. Thus, well parameterized but unknown forward models in 3D imaging systems such as stereo rang imaging systems can not be fitted into the framework mentioned above. 2). The second limitation lies in the regularization term Φ. The form of the regularization term determines the *a priori* assumption or the preference enforced in the structure of the reconstruction $\hat{f}$. For instance, the Total Variance regularizer enforces the smoothness in the reconstructed object $\hat{f}$, while the L_1 norm regularizer enforces sparsity approximately. The selection of regularization plays a vital role in the reconstruction but is a subjective choice and may not accurately reflect the *a priori* of the true object f. In this dissertation, we propose 1) a novel framework to apply computational imaging on well parameterized unknown forward models[12, 18, 19, 20], namely the range imaging systems in Chapter 2; 2) a data-driven estimation of the object distribution and verify it in an adaptive sensing task[21] in Chapter 3 to overcome these limitations.

The Dissertation is organized as followed. In Chapter 2, we present the temporal com-

pressive range imaging systems which include unknown, well-parameterized forward models. In Chapter 3, we present the adaptive sensing framework which captures the object distribution with generative neural networks. We conclude the dissertation with Chapter 4.

CHAPTER 2: COMPUTATIONAL IMAGING FOR HIGH-SPEED RANGE IMAGING

2.1 Range Imaging

Range imaging systems, which collect three-dimensional spatial information of the object surface, have a wide range of applications in medical procedures[22, 23], archaeological landscape mapping[24], industrial metrology, 3D printing, tracking, and vehicle navigation[25]. Range imaging systems can be roughly categorized into either passive or active sensing methods [26]. Passive sensing methods are cost-effective and simple in implementation while active illumination has more degrees of freedom in controlling the properties of the light source, such as wavelength, polarization, coherence, temporal profile, etc.

2.2 Passive Computational Range Imaging

Passive imaging system is robust and cost efficient, making high-speed 3D imaging more accessible. We demonstrate a compressive stereo imaging setup as an attempt to implement a passive high-speed depth sensing system. Bearing the system cost in mind, we engineered an asymmetric stereo imaging system that includes the high-speed modulator in only one of the optical paths, while keeping the other optical path unmodified, simply a low-frame-rate camera to capture a low-frame-rate blurry scene. To reconstruct the high-speed 3D scene, a general framework is proposed to estimate the depth and intensity information from the two measurements. The major challenge is to estimate the depth from the two asymmetric optical paths and in order to address this, we develop a two-step algorithm, in which the first step recovers the high-speed scene from the modulated optical path and the second step extracts the depth of the scene by employing the information from both measurements. Stereo imaging system estimates 3D scene from measurements taken from left and right views. Two pixels in these measurements are correspondent, if they refer to the

same element in the scene. In rectified epipolar geometry, corresponding pixels are on the same row, and the location difference of these pixels is called disparity. The depth of an object in the scene can be inferred from the disparity between these two measurements.

2.2.1 Forward Model

Figure 2.1 depicts our compressive stereo imaging system. Both left-view and right-view measurements are synchronized and sampled at a low frame-rate. The left-view measurement I_L captures the summation of the high-speed scene within exposure time. On the right-view optical path, the high-speed scene F_{RH} is modulated by N high-speed pseudo-random patterns $M(i,j,n)$ during the exposure time, where (i,j) is the piexel coordinates of the right sensor. The modulated scene is then relayed to Camera 2 forming the right-view measurement I_R. Considering the stereo measurements of each view has $N_x \times N_y$ pixels, the pixel (i,j) can be expressed as:

$$I_R(i,j) = \sum_{n=1}^{N} F_{RH}(i,j,n)M(i,j,n) \tag{2.1}$$

$$I_L(i,j) = \sum_{n=1}^{N} F_{LH}(i,j,n) \tag{2.2}$$

where $i,...,N_x, j,...,N_y$. F_{LH}, F_{RH} are the high-speed scene from left-view and right-view, respectively. As mentioned above, in the stereo imaging system, the high-speed depth information lies in the correspondence between F_{RH} and F_{LH}. However, neither F_{RH} nor F_{LH} can we measure directly since the frame-rate of F_{RH} and F_{LH} exceeds that of the cameras. Our contribution is to estimate the high-speed scene from low frame-rate measurements I_L and I_R.

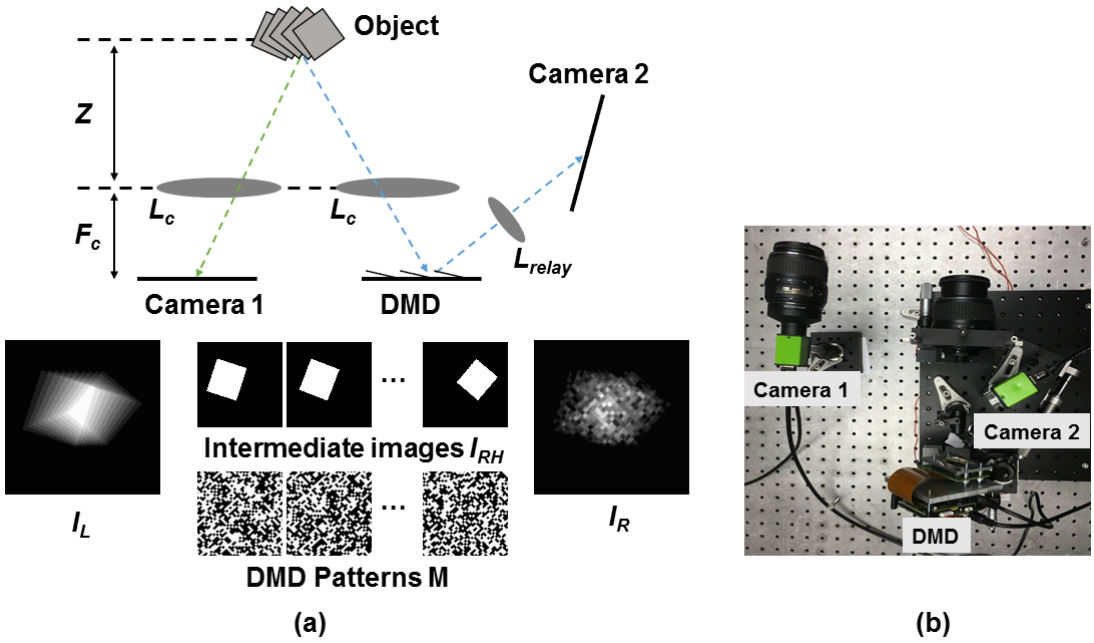

Figure 2.1: System schematic (a) On the left-view optical path, Camera 1 records low-speed measurement IL. On the right-view optical path, high-speed right-view scene F_{RH} are encoded by the DMD with N distinct patterns M. The coded right-view scene is then relayed by lens Lrelay and recorded by Camera 2 within the exposure time to form the right-view measurement IR. (b) A photo of the setup

2.2.2 Reconstruction

After capturing the measurements shown in Equation 2.1-2.2, we aim to estimate the high-speed scene as well as the depth. Let $F(i,j,k,n)$ denote the high-speed 3D scene that we are interested, where i,j symbolize the spatial indices, k signifies the depth information and n is the high-speed frame index. The relation between the left- and right-view high-speed scenes can be expressed as where H is a transformation matrix depending on the depth. Since we only need to estimate F_{RH} and H to obtain the high-speed 3D scene, the reconstruction problem can be formulated as

$$(\hat{F}_{RH},\hat{T}) = \underset{F_{RH},T}{\arg\min} \left\| I_R - \sum_{n=1}^{N} F_{RH}\cdot M \right\| + \left\| I_L - \alpha \sum_{n=1}^{N} F_{RH}\cdot H \right\| \tag{2.3}$$

where α is used to compensate the intensity difference between the two optical paths. Unfortunately, Equation 2.3 is ill-posed. We add prior knowledge on F_{RH} as regularizers to solve Equation 2.3. This leads to

$$(\hat{F}_{RH}, \hat{H}) = \underset{F_{RH},T}{\arg\min} \left\| I_R - \sum_{n=1}^{N} F_{RH} \cdot M \right\| + \lambda \Phi(F_{RH}) + \left\| I_L - \alpha \sum_{n=1}^{N} F_{RH} \cdot H \right\| + \kappa \Omega(H) \tag{2.4}$$

where Φ and λ are the regularizer and weight for F_{RH}, Ω and κ are the regularizer and weight for H respectively. The two arguments are coupled since third term in Equation 2.4 is a coupled term of F_{RH} and H. In this paper, we ignore the impact of third term in estimating F_{RH} and propose a two-step algorithm to solve the high-speed video (F_{RH}) first, and then estimate the high-speed depth maps (H). This approximation works well in practice as our results shown. In the following, we consider F_{LH},F_{RH} in the rectified epipolar geometry, and therefore T can be explicitly represented by the disparity shown in Equation 2.6. The right-view high-speed scene F_{RH} is first reconstructed from the snapshot I_R. Secondly, we solve the correspondence problem between single left-view measurement I_L and N-frame high-speed right-view scene F_{RH}.

In the first step, we reconstruct high-speed scene F_{RH} from the snapshot I_R in Equation 2.1. This is a video compressive sensing inversion problem. Since the DMD multiplexes N frames and collapses into one measurement, this inversion problem is ill-posed. Here, we use the iterative reconstruction algorithm TwIST to solve the optimization problem [13]

$$\hat{F}_{RH} = \underset{F_{RH}}{\arg\min} \left\| I_R - \sum_{n=1}^{N} F_{RH} \cdot M \right\| + \lambda \Phi(F_{RH}) \tag{2.5}$$

The TV regularizer is employed to promote piece-wise smoothness in estimates, since natural scenes are usually sparse in spatial gradients [27]. After this step, we obtain N high-speed frames F_{RH} from right-view optical path, while only a single blurry measurement I_L is available from the left-view optical path.

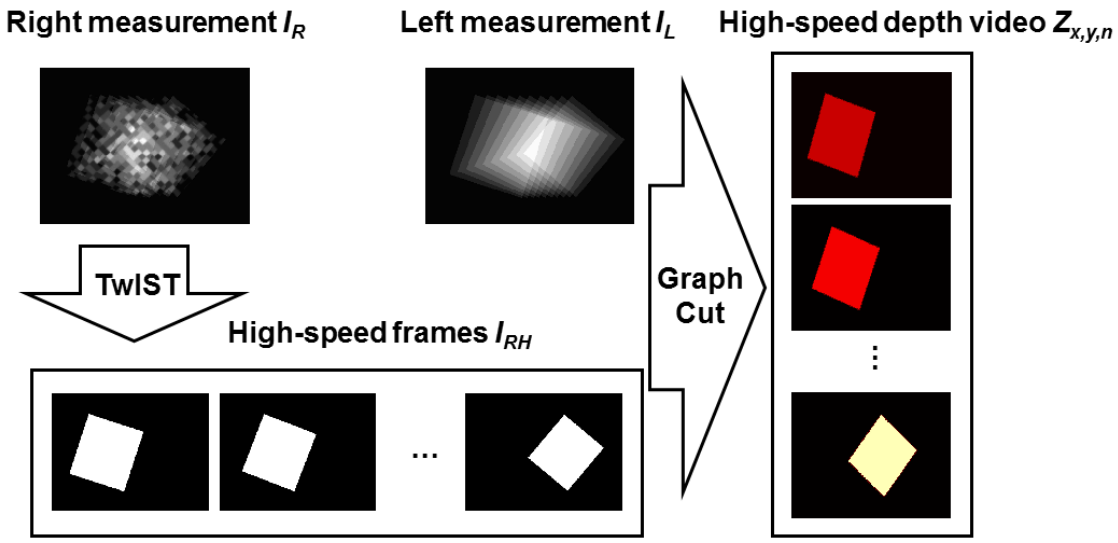

Figure 2.2: The flow chart of the reconstruction algorithm. The high-speed scene F_{RH} is reconstructed from the modulated measurement I_R using video compressive sensing inversion algorithm, TwIST. Then, the high-speed depth maps are estimated from I_L and F_{RH} by our one-to-N correspondence algorithm based on Graph Cut.

The second step is to estimate correspondence between I_L and F_{RH}. Although various correspondence algorithms [28, 29] exist to estimate the disparity map, they aim to find the one-to-one correspondence between two measurements. In our system, I_L does not correspond to any single frame in F_{RH} but to all N high-speed frames. To explicitly represent T in Equation 2.4, let $D(i,j,n)$ denote the disparity of pixel (i,j) in n^{th} frame between the right-view high-speed scene F_{RH} and the corresponding left-view low-speed measurement I_L. Considering the radiometric difference, the un-occluded pixels in I_L can be represented as

$$I_L(i,j) = \frac{1}{\alpha}\sum_{n=1}^{N} F_{RH}(i+D(i,j,n),j,n) \tag{2.6}$$

where α is the radiometric ratio between the two paths. In the experiments, we calibrate this α at the beginning. Therefore, for simplicity, the following discussion will set $\alpha = 1$. The depth Z(i, j,

n) can be calculated by

$$Z(i,j,n) = \frac{f_c b}{D(i,j,n)} \tag{2.7}$$

where f_c is the focal length of the camera lens, b is the baseline length; f_c and b can be obtained by calibration [30]. Computing the one-to-N correspondence between I_L and N-frame video F_{RH} is the main challenge thus the vital ingredient of our algorithm. We formulate this as an energy minimization problem, and propose a correspondence algorithm based on Graph Cut [31]. More specifically, we estimate the disparity by

$$\hat{D} = \arg\min_{D} E_{data}(D) + E_{regularizer}(D) \tag{2.8}$$

where E_{data} and $E_{regularizer}$ denote the data term and the regularization term of the energy function, respectively. In our system, $E_{data(D)}$ is used to measure the similarity of the corresponding pixels according to Equation 2.6. Since the measurements in different perspectives are rectified, corresponding pixels are on the same row. Employing the absolute difference as metric, $E_{data}(D)$ is defined by a one-to-N assignment

$$E_{data}(D) = \sum_{i,j} \left| I_L(i,j) - \sum_{n} F_{RH}(i + D(i,j,n), j, n) \right| \tag{2.9}$$

In order to engineer diverse problems during the matching process in stereo imaging systems, the regularization term $E_{regularizer}$ is composed of three terms:

$$E_{regularizer} = E_{occlusion} + E_{uniqueness} + E_{smoothness} \tag{2.10}$$

$E_{occlusion}$ is a penalty to the occluded pixels,

$$E_{occlusion} = K_{occ} N_{occ} \tag{2.11}$$

where K_{occ} is a constant and N_{occ} is the number of matchings labeled as occluded. $E_{uniqueness}$ is to enforce the uniqueness of the matching in D,

$$E_{uniqueness} = K_{uniqueness} T(D) \tag{2.12}$$

where $K_{uniqueness}$ is a large constant, $T(D)$ is used to detect the uniqueness of D. $T(D) = 1$ if any pixel in I_L and F_{RH} is involved in more than one assignments, otherwise $T(D) = 0$. $E_{smoothness}$ promotes the piece-wise smoothness in D. For two adjacent pixels p, s in I_L, if p and s have different disparities, we give a penalty V. Let D_p and D_s denote the disparity at pixel p and s, the smoothness term is defined as

$$E_{smoothness} = \sum_{s,p \in \aleph, D_s \neq D_p} V \tag{2.13}$$

(13) where $\aleph$ is a neighborhood set and V is a constant. In our algorithm, a small constant V_1 is used when $|I_L(p) - I_L(s)|$ is above a pre-defined threshold, otherwise a larger constant V_2 is used. The underlying rationale is to match the depth jump with the intensity jump. By using these graph representable energy terms and an appropriate definition of the smoothness term, we can find a strong local minimum of the problem in Equation 2.8 via Graph Cut [31, 32]. Empirically, we have found that this definition has led to a strong local minimum of the problem in Equation 2.6, which is sufficient for our applications.

The depth resolution can be derived from Equation 2.7. By taking derivative of both sides of Equation 2.7, we have $dZ = \frac{f_c b}{D^2} dD$ By substituting D with $\frac{f_c b}{Z}$ in the denominator, the depth resolution can be expressed as

$$dZ = \frac{Z^2}{f_c b} dD \tag{2.14}$$

where dD is the spatial resolution of the images. In our setup, $f_c = 26mm$, $b = 174mm$, the scene is around 1.6 m away from cameras. The pixel size of the camera $dD = 5.5\mu m$. The calculated theoretical depth resolution is around 3 mm. In our experiments, for computational efficiency, we

down-sampled reconstructed F_{RH} and left-view measurement I_L by 8, and estimate the high-speed depth map based on the down-sampled images. Therefore, the theoretical depth resolution in our experimental results is around 2.4 cm.

2.2.3 Experimental Results

We built our prototype demonstrated in Figure 2.1b. The same camera lens (Nikon, 18-55mm) and camera (JAI, GO 5000M) are used on both optical paths. The cameras are triggered and synchronized by the data acquisition board (NI, USB6353). On the left-view optical path, the camera is placed on the back focal plane of the camera lens. On the right-view optical path, the high-speed scene is modulated by a DMD (Vialux, DLP7000), and then relayed to the camera by the relay lens (Edmund Optics, 30mm, f/8). The pitch of the DMD is 13.7 μm with fill factor of 0.92. The DMD is working in 3 binning mode. To extend the working distance, we utilize another relay lens to relay the intermediate images (on the back focal plane of the camera lens) to the DMD.

In the first example, the cameras operate at 30fps. The compression ratio N equals to 10. The left and right measurements are shown in Figure 2.3a. The different directions of motion blurs on I_L and I_R indicate the varying depth of the ball within the exposure time. This is a challenging problem for any existing stereo imaging systems and correspondence algorithms: estimating the varying depth from motion blur without the knowledge of the shape of object. By contrast, we address this using our proposed reconstruction framework.

Following the flow-chart of our algorithm, we first reconstruct 10 high-speed frames from the right measurement I_R and the calibrated DMD patterns M, with results shown in Figure 2.3b. After this, we send these 10 frames along with the left measurement to the correspondence algorithm we have built in Equation 2.6. The outputs of the algorithm are 10-frame depths as shown in Figure 2.3c. Considering N_x columns in the measurement, N_x^N different disparities are possible in one-to-N matching while only N_x possible disparities in one-to-one matching. The size of

the searching space will be a challenge when N and N_x become large. In this example, $N_x = 125$ after down-sampling. Under the linear motion assumption in a short duration, we can decrease the searching space size from N_x^N to $N_x \cdot N_s$, where N_s is the number of possible velocities along z axis that can be detected. The frame-rate of the reconstructed video is 300fps which is 10 times as that of the camera. The overlay plot of the depth map in Figure 2.3d demonstrates the estimated motion of the ball. The depth increment of adjacent frames is around 23 mm which approximately corresponds to the theoretical disparity of one pixel, i.e., 2.4 cm. The estimated average velocity of the ball along z axis is 6.7 m/s.

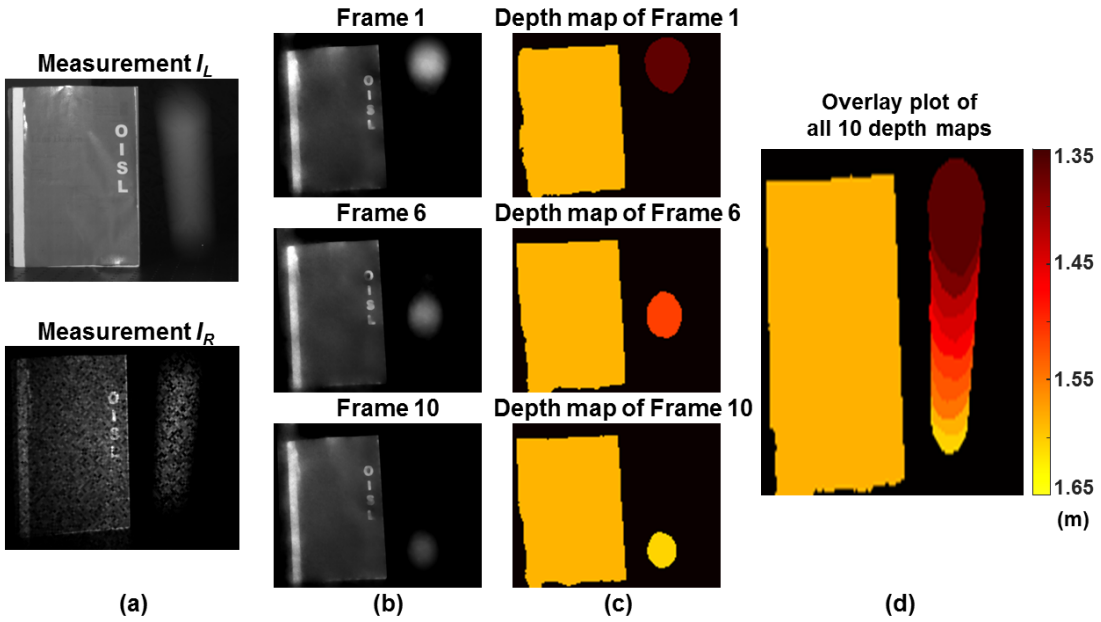

Figure 2.3: Reconstruction of a backward-moving ball. (a) Measurements from two optical paths in our system. The different traces of the motion blurs indicate the varying depth of the moving ball. (b) 1^{st}, 6^{th} and 10^{th} fames of reconstructed high-speed video from single measurement I_R. (c) 1^{st}, 6^{th} and 10^{th} fames of reconstructed high-speed depth map. (d) An overlay plot of 10 depth maps within a single exposure. The gradient of the color implies that the ball is moving away from our imaging system.

In the second example, we test our system with scenes containing more complicated motion by the cameras operated at a higher frame rate. The scene consists of a stationary box with

letters “UCF”, a fast-moving triangular shuriken and a moving rectangular shuriken. The camera is operating at 80fps, while the compression ratio is still 10. Thus, the expected frame-rate of the reconstructed video is 800fps. As shown in Figure 2.4, the motion blur in the measurement indicates that the motions of the shurikens are mixture of transformation and rotation. Similar to the first example, we first reconstruct the high-speed video frames, now at 800fps, shown in the top-right of Figure 2.4. Then our correspondence algorithm provides the depth maps for these 10 frames. An 800fps 3D video is reconstructed from 80fps measurements (See Visualization 2). The rectangular shuriken rotated 30 degrees while the triangular shuriken rotates 20 degrees within the exposure time.

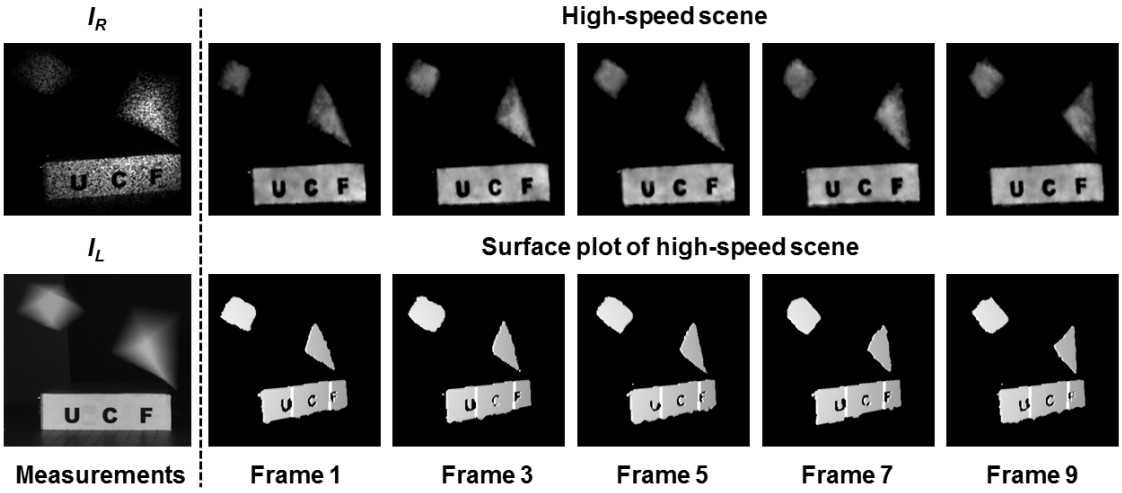

Figure 2.4: Reconstruction of an 800fps high-speed scene with two flying “shurikens”. (a) Stereo imaging measurements. (b) Selected frames of reconstructed 800fps video. (c) 3D rendering of the high-speed scene. The rectangular shuriken rotated about 30° within the exposure time, and the triangular shuriken rotated about 20°

In summary, we have reported a high-speed compressive stereo imaging system, and a two-step inverse algorithm. We have reconstructed a 3D video at 800 fps from coded stereo measurements at 80 fps. Our system exploits the correlations of temporal, spatial and depth channels of the information in passive depth sensing.

2.3 Active Computational Range Imaging

2.3.1 Forward Model

We consider the high-speed three-dimensional (3D) video which can be modeled as an intensity function $f(x,y,t)$ and a depth map $Z(x,y,t)$. The camera integrates the high-speed frames $f(x,y,t)$ within the integration time T into one measurement. To discern the range of the scene and retrieve the high-speed temporal frames, we have implemented the two-fold modulation using the structured illumination. In our setup shown in Figure 2.5, we use a projector to project high-speed pseudo-random binary masks onto the 3D scene. We establish the coordinates with the origin located at the lens in the projector. Let $h^*(x,y,t)$ denote the three dimensional high-speed masks imposed on the scene $Z(x,y,t)$. The measurement $g(x',y')$ can be expressed as

$$g(x',y') = \int_{t=0}^{T} f(\frac{z+l_0}{f_c}x', \frac{z+l_0}{f_c}y', t) \cdot h^*(\frac{z+l_0}{f_c}x', \frac{z+l_0}{f_c}y', t) \tag{2.15}$$

where f_c is the focal length of the camera lens, l_0 is the distance between projector and camera along z axis. In temporal domain, the camera integration time T limits the passband of the temporal information acquired by the camera. However, the high-speed temporal masks $h^*(x,y,t)$ alias the higher frequency components of scene $f(x,y,t)$ into the passband of the camera. Therefore, we have a chance to reconstruct the high-speed frames through the inversion algorithms proposed in Section 3.

One key contribution of our work is to reconstruct the depth information of the scene as well as the temporal super resolution. Different from the previous method that modulates the range of the scene with varied blur kernels [33], we take advantage of different scales and shift of the masks. Let $h^z(x,y,t)$ denotes the ideal projected images of the original masks $h^0(x,y,t)$ projected to the range z without any objects (e.g., a uniform white background). Considering the lateral offset d (in x axis) between the camera and the projector as shown in Figure 2.6, the ideal projected masks

can be expressed as

$$h^z(x,y,t) = h^0(\frac{f_p}{z}x + \frac{f_p}{z}d, \frac{f_p}{z}y, t) \tag{2.16}$$

where f_p is the focal length of the lens of the projector. It is worth noting that h^z is a plane located at range z. With this structured illumination, we modulate the range with scaling factors $\frac{f_p}{z}$ and the shift $\frac{f_p d}{z}$ which are both the functions of range z. The ideal projected masks h^z can be obtained by calibrations or simulations with the parameters fp, d, and the origin masks h^0. Our goal is to estimate $f(x,y,t)$ as well as the depth map $Z(x,y,t)$, given $g(x',y')$ with prior of h^z.

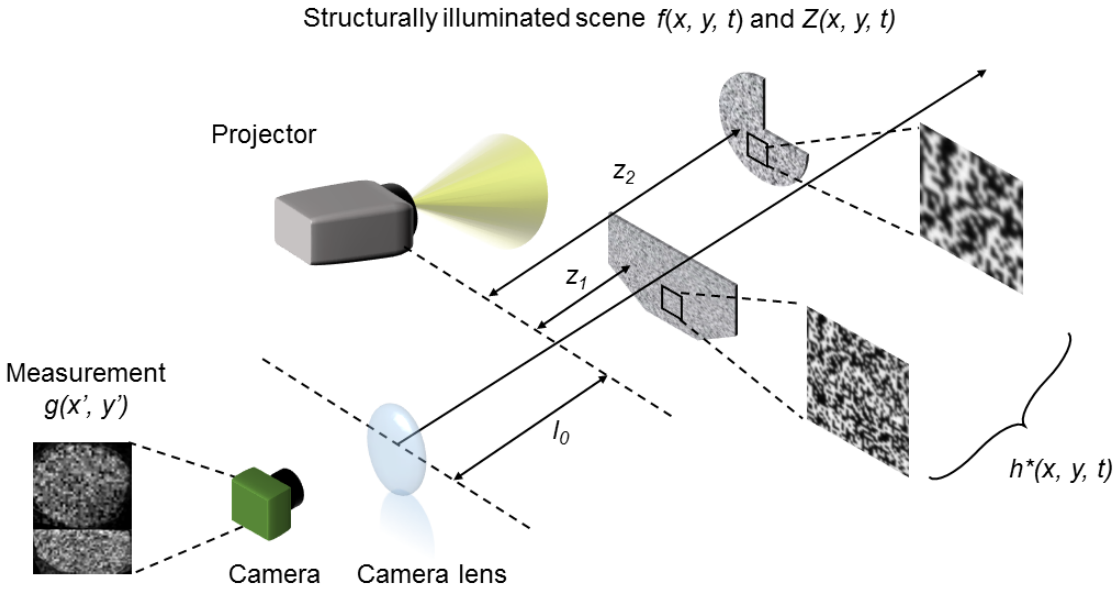

Figure 2.5: The forward model of our system. The projector projects high-speed masks onto the scene. The scene is modulated by $h^*(x,y,t)$, the variants of the original mask at different ranges. The camera integrates the modulated high-speed frames into one measurement.

2.3.2 Reconstruction

We can discretize high-speed video f, projected masks h^*, and measurements g. Let $F_k \in R^{N_x \times N_y}$ denote the k^{th} discretized frame of the scene. For each pixel $(m,n), m = 1,...,N_x; n = 1,...,N_y$, we

have

$$g_{m,n} = \sum_{k=1}^{N_T} h_k^*(m,n) \cdot f_k(m,n) \tag{2.17}$$

where we consider N_T discretized high-speed video frames within the integration time T. Let f_k be the vectorized form of F_k. The vectorized form of the measurement can be expressed as

$$\mathbf{g} = \mathbf{H}^*[\mathbf{f_1}...\mathbf{f_{N_T}}]^T \tag{2.18}$$

The inverse problem can now be formulated as

$$(\hat{\mathbf{f}}, \hat{\mathbf{H}}^*) = \arg\min_{\mathbf{f},\mathbf{H}^*} \|\mathbf{g} - \mathbf{H}^*\mathbf{f}\| + \tau R(\mathbf{f}) \tag{2.19}$$

$\mathbf{H}^*$ is the sensing matrix corresponding to the projected masks h^*, and $R(\mathbf{f})$ denotes a regularizer which can be used to impose the sparsity of the signal in the basis such as the wavelet, the discrete cosine transformations (DCT) or the total variation (TV) operator.The regularizer penalizes characteristics of the estimated f that would result in poor reconstruction. τ is the Lagrange parameter balancing the measurement error and the regularizer. There are two parameters to estimate in Equation 2.4, which is non-convex. However, given one, the other one can be solved via existing algorithms[34, 35]. In the following, we solve the problem by alternatively estimate one with the other one fixed. Although we cannot directly obtain the projected masks h^*, we know that h^* is a combination of different portions of the ideal masks h^z at different z. Considering the spatial information as well as the range resolution of the reconstructed results, we apply local window to crop the measurement into small block, and slide the window in both horizontal and vertical directions with sub-block increments δ to obtain a sequence of blocks [19] and estimate a range $\hat{z_{block}}$ for each block. Let N_z and $\{\mathbf{H}^i | i = 1,...,N_z\}$ denote the number of discretized ranges and the ideal sensing matrix corresponding to the ideal projected masks $h^z(x,y,t)$ at i^th discretized range, respectively. For each block, we can enumerate the ideal sensing matrices $\mathbf{H^i}$ and obtain N_z

candidates of the reconstructed fraction of the scene $\hat{\mathbf{f}}_{\mathbf{block}}$ as

$$\hat{\mathbf{f}}^{i}_{block} = \arg\min_{\mathbf{f}_{block}} \left\| \mathbf{g} - \mathbf{H}^{i}\mathbf{f}_{block} \right\| + \tau R(\mathbf{f}_{block}) \tag{2.20}$$

where $i = 1, ..., N_z$. Equation (20) can be solved by a commonly used compressive sensing inversion algorithms, for example, the TV based optimization algorithms[34] or the Bayesian algorithms [36]. The second step is to select the one fitting the measurement best,

$$\hat{z}_{block} = z_{i^*}, i^* = \arg\min_{i} \left\| \mathbf{g}_{block} - \mathbf{H}^{i}\hat{\mathbf{f}}^{i}_{block} \right\| \tag{2.21}$$

where z_{i^*} is the i^{th} range. Empirically, we have found adding the regularizer on each block can provide better results.

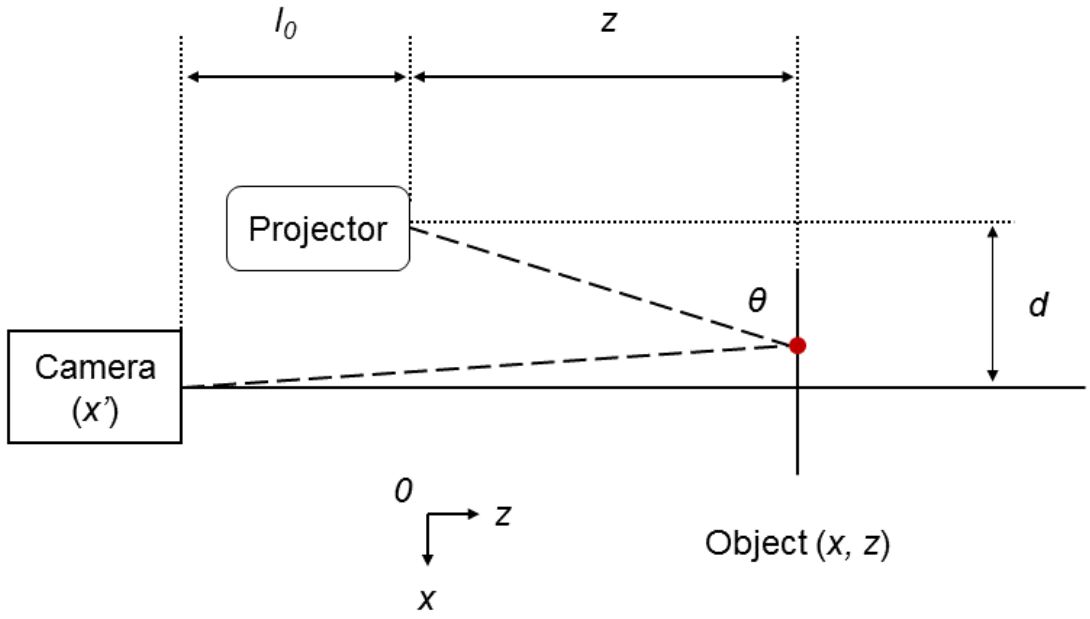

Figure 2.6: Schematics of our system. The lateral offset of projector and camera is d. The distance between projector and camera along the z axis is l_0. θ denotes the projection angle. The focal lengths of the camera and the projector are not specified in the figure for concision.

We assume that the depth map $Z(x,y,t)$ does not change within one measurement, thus we just enumerate N_z possible sensing matrices here. For the scenarios without this assumption, the number of possible sensing matrices is $N_T^{N_Z}$, which grows exponentially with the compression rate N_T. After we select the best range for each block, we can get the corresponding reconstruction for each block. This reconstruction is based on the inferred scale and shift of the original mask h^0 and the measurement g. The final results can be obtained by fusing these reconstruction blocks. However, due to the spatial correlation of different blocks. After getting the range for each pixel, we can obtain the correct mask for each pixel. Equation 2.19 can now be solved via the video compressive sensing algorithms, which considered both the global and local information.

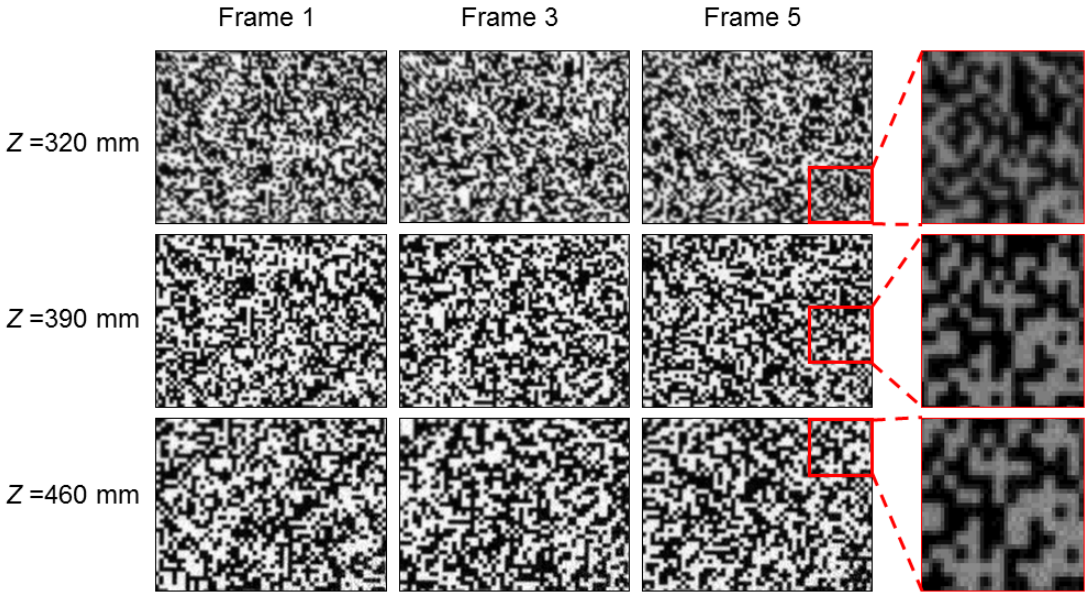

Figure 2.7: Calibrated masks at different ranges. Same patterns of the 5^{th} mask projected at different ranges are highlighted with the red boxes. The shift of the red boxes indicates one modulation on the range. The scales of the zoom in patterns is the other modulation on the range.

2.3.3 Experimental Results

The schematic is shown in the Figure 2.6. The projector consists of a microscope lamp (Nikon, D-LH), a Digital Micromirror Device (Vialux, DLP4100) and a camera lens (Nikon, 18-55mm). A Digital Micromirror Device (DMD) is an array of the highly reflective aluminum micromirrors. Each pixel is an electro-mechanical element in which there are two stable micromirror states ($\pm 12^{\circ}$). The DMD with an adequate optical element can be used as a high-speed projector. The maximum frame rate of the DMD is 22.7 kHz which is much faster than the camera. To modulate the high-speed frames of the scene, the projector is used to project pseudo-random binary masks onto the scene at 1000 frames per second (fps) while the camera is acquiring the measurement at 200 fps. Objects at different ranges are modulated by different parts of the mask with different shift and magnifications. The shift is induced by the separation of the optical axis of the projector and the camera. It is worth noting that the feature size of masks acquired by the camera depends not only on the distances to the projector L_p but also on the distance to the camera L_c. The corresponding pixel size is proportional to the ratio of $\frac{L_p}{L_c}$. If the baseline of the camera and the projector are the same, the pattern feature on the camera would be the same regardless the range of the objects, in which case the only modulation on the range is the shift. To better discern the pattern at different ranges, we place the camera and the projector at different locations on the z axes. The modulated objects are imaged by a camera lens (Nikon, 18-55mm) onto the camera (JAI, GO5000M). The separation of the axes of the camera and the projector is $d = 135$ mm. The focal length of the camera lens is 50 mm. To ensure the coding process remains time-invariant, we write the pseudo-random patterns into the memory of the DMD prior to the display, and then use an NI board (NI, USB 6353) to synchronize the camera and the DMD in the projector. The NI board generates a pulse to start the DMD display, then generates a 200 Hz square wave to trigger the camera. There is a fixed delay between the DMD and the camera control signals to ensure the synchronization of these two. We use an active area of 296×325 detector pixels to account for the

128x96-pixel pseudo-random patterns with additional zero-padding. To implement the algorithm, we need to calibrate the ideal sensing matrix H^i mentioned in Section 3. Firstly, we record each pseudo-random mask projected on a white board located at i^{th} range z_i. Secondly, we record the image from an all-on state DMD to correct the nonuniformity of the illumination. Thirdly, we record the image of an all-off state DMD for subtraction of the background. After the corrections to the images of the masks, we can extract H^i. At last, we repeat the same calibration procedures at discrete steps of 10 mm on the z axis. The whiteboard is translated by two motor-driven translation stages (Thorlabs, NRT100 and Newport, LTA-HS) which give us a 150 mm travel range in total with stable and repeatable translation. Some calibrated masks at different ranges are shown in Figure 2.7.

We now use the camera to capture a experimental high-speed 3D scene and reconstruct it. The setup configuration is shown in Figure 2.6. Before acquiring the data, we calibrate the camera with checker board pattern to correct the aberrations. We place two objects at different ranges, one is a stationary white board S1 ($68mm \times 28mm$) which is located at $z_1 = 320mm$ away from the projector and $l_0 + z_1 = 1.63m$ away from the camera. The other object is a circular board S_2 (36 mm in diameter) which is fixed on a fan with a notch rotating at 30 rounds per second. A black square tape ($18mm \times 18mm$) is added to the surface of S_2, serving as a reference mark. S_2 is placed at 360 mm away from the projector. The camera is operating at 200 fps, and we reconstruct five frames for each measurement. Thereby, we can retrieve high-speed video frames at 1000 fps with depth maps. The scene is shown in Figure 2.5 and we show one measurement and reconstruction results in Figure 2.8. We set the block size to 16×16 pixels, and the increment of the blocks δ is 8 pixels. It can be observed that not only can we reconstruct the scene at 1000 fps but also provide the corresponding depth map. Our desktop with 8 gigabytes memory completes the reconstruction for one measurement within 48 seconds. As our algorithm is based on blocks and each block is reconstructed independently, parallel computing is ready to be used with a GPU. The computing time can be reduced by orders of magnitude. There are some artifacts on the boundary

of both objects because the resolution of the depth map depends on the size of the block and the overlapping between the adjacent ones as mentioned in Section 3. Whenever there is a steep slope in the block, the artifacts will appear. These can be improved by using blocks with more overlap with compromised computation time.

2.4 Limitation and Outlook

We study the computational imaging applications in high-speed stereo vision systems. In the first part, we have reported a temporal compressive passive stereo imaging system, and a two-step inverse algorithm. We have reconstructed a 3D video at 800 fps from coded stereo measurements at 80 fps. Our system exploits the correlations of temporal, spatial and depth channels of the information in passive depth sensing. From the hardware perspective, the temporal limit of our system is the DMD refreshing rate. A faster modulation can lead to an even faster frame rate. However, this does not mean the frame rate can be increased in this fashion. The faster frame rate will also reduce the signal to noise ratio (SNR) of the measurement, which is detrimental to the reconstructed images. The depth resolution of system is limited by the triangulation geometry of the stereo imaging system, which is on the order of centimeter. Specifically, the depth resolution depends on the spatial resolution of the images, the distance between the scene and the camera, the focal length, and the baseline distance. In the compressive sensing system, the spatial resolution is also affected by the matching between the pixel size of the camera and the feature size of modulation pattern on DMD. On one hand, large feature size would result in low spatial resolution and larger errors in depth triangulation; on the other hand, smaller feature size could result in a poor calibration, which would inversely affect the reconstruction. Here we would like to mention the option of using two spatial modulators, e.g. two DMDs, in both optical paths. The reconstruction can be simply divided to 1) recovering the high-speed videos from both paths and 2) calculating the corresponding depth maps. In addition to the obvious advantages of lower power consumption and

lower system cost of our imaging setup, leaving one optical path unmodified maximizes the light collection efficiency of the stereo imaging system. Our recent results show that this light-collection improvement could lead to a superior reconstruction in the temporal compressive system with two identical channels.

We envision an integrated reconstruction frame work that merges the current two reconstruction steps. The depth estimation could be used to transform the left-view measurement as side information to improve the high-speed reconstruction of the right-view. An iterative process of updating the right-view reconstruction and depth map could thus be implemented. Different from active illumination system which is sensitive to the ambient light, the reported method is suitable for passive depth sensing system and can be directly implemented using a color camera, making the RGBD sensing system more accessible

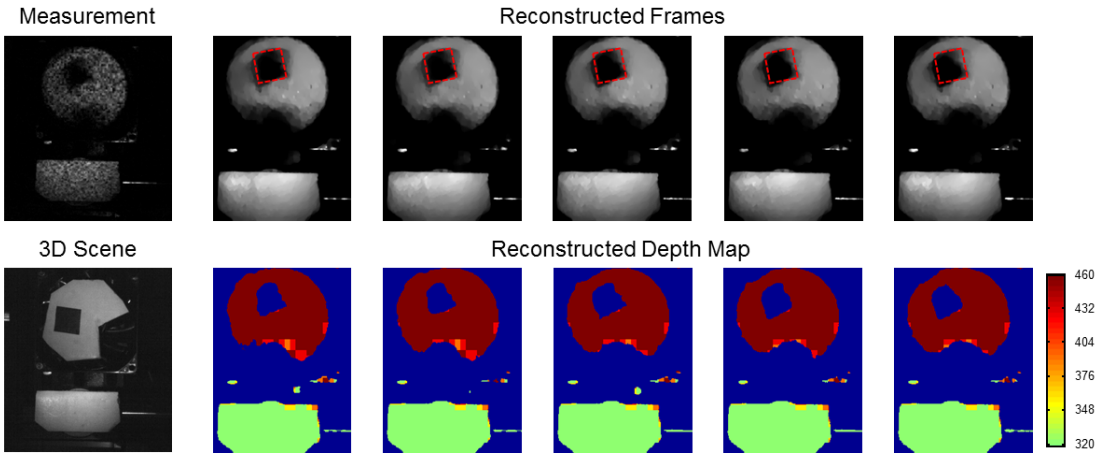

Figure 2.8: The top row presents one frame of a 200 fps measurement and the reconstructed 1000 fps reflectance frames. The object is a disk rotating at 30 rounds per second. The red box indicates the position of the black tape in the first frame. The bottom row shows the ground truth of the 3D scene and the reconstructed depth map. A video of 1000-fps frames has been reconstructed from 200-fps measurement

In the second part, we describe a novel temporal compressive active range imaging system

that encodes both range and high-speed temporal information of a scene with a series of binary random patterns projected by a high-speed DMD. We have demonstrated the imaging principle by reconstructing a fast-varying scene. To solve the inverse problem, a block-wise alternating algorithm has been developed to reconstruct high-speed temporal and range information. A 1000-fps video of reflectance intensity with depth map is reconstructed from 200-fps measurements. In this work, we describe a novel range imaging system that encodes both range and high-speed temporal information of a scene with a series of binary random patterns projected by a high-speed DMD. To solve the inverse problem, a block-wise alternating algorithm has been developed to reconstruct high-speed temporal and range information. In our reconstruction algorithm, the range resolution is determined by the sensitivity of the reconstruction error, which is inversely related to the correlation of projected patterns at different ranges. The simulation has demonstrated a range resolution better than 3.2 mm, which is in agreement with the range resolution calculated from the triangulation method. To improve the range resolution, a finer projection pattern and further optimization of the system geometry is needed. Our system is simple to implement and has potential applications in high-speed range imaging.

CHAPTER 3: COMPUTATIONAL IMAGING IN ADAPTIVE SENSING

3.1 Adaptive Sensing and Sensor Placement

Optimal sensor placement achieves the minimal cost of sensors while obtaining the prespecified objectives. In this work, we propose a framework for sensor placement to maximize the information gain called Two-step Uncertainty Network(TUN). Experiments on the synthetic data show that TUN outperforms the random sampling strategy and Gaussian Process-based strategy consistently.

Sensor placement is widely studied in the areas of environment monitoring [37, 38], structural health monitoring[39] , security screening[40], and adaptive computed tomography[41]. The optimal sensor placement maximizes the objectives with minimal cost of sensors. Given the model that maps each possible set of sensor locations to the objectives, the optimal sensor placement can be formulated as an optimization problem. However, the optimization is shown to be NP-hard[42]. Thus, approximate greedy algorithms of sequential sensor placement are proposed and then proved to be near optimal under the assumptions that the criterion are monotone and submodular[43].

The diagram of a sequential sensor placement is shown in Figure 3.1a with the black arrows. The agent inquires at a feasible location to the physical model in each step and obtains the corresponding measurement. The obtained observations are used to make inference for specific tasks. For instance, in security screening tasks the observations are used to predict the distribution of the object's label. To make an accurate inference, the agent often optimizes the information gain in each step with respect to the feasible location. The corresponding objective is mutual information which is approved to give near optimal approximations in sequential sensing[44, 43].

To optimize the objective, a model that estimates the potential information gain at each possible location is necessary. The most generic method to model the unknown spatial phenomenon is Gaussian Process(GP) which incorporates the knowledge of observations and predicts the uncertainty at the un-observed locations. However, the Gaussian model assumption in GP does not

perform well on high dimensional data e.g. images as generative models. In addition, GP inherently adapts the assumptions that the uncertainty at the un-observed locations is independent of the obtained measurements, making the GP based sequential sensing an open-loop control[45]. An alternative approach to this problem arises recently is Reinforcement Learning(RL)[46]. However, the performance in RL is found to be of large variance and difficult to reproduce[47].

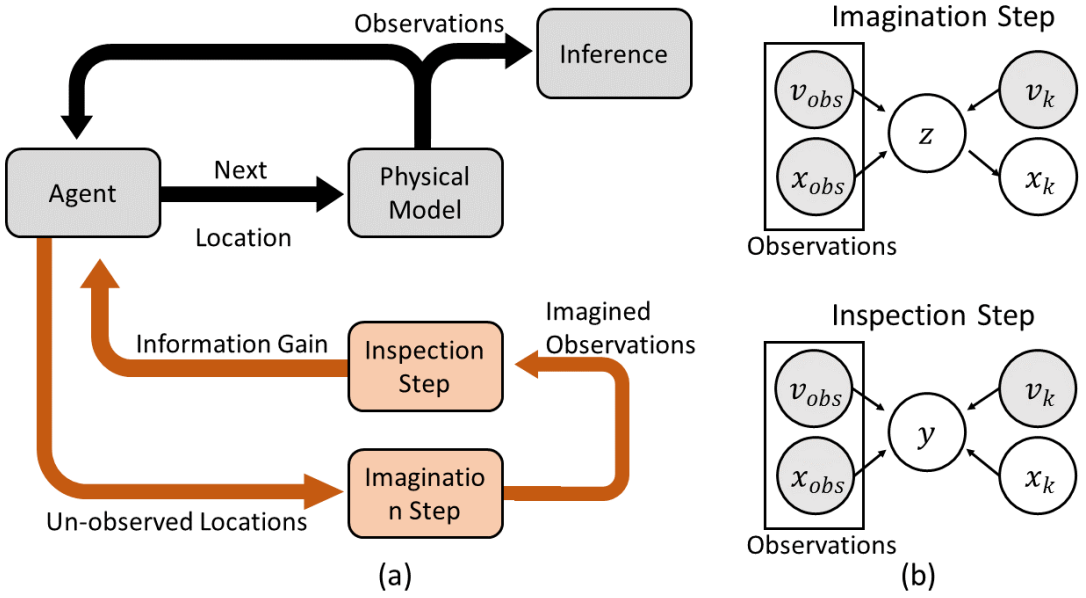

Figure 3.1: (a) The diagram of sensor placement (black arrows) and TUN(red arrows). TUN consists of two steps: imagination step and inspection step to generate the possible measurements and evaluate the task-specific information gain at un-observed locations. (b) Graphical models of the two steps in TUN. Instances of measurement x_k at un-observed locations are generated in the imagination step. Then those generated measurements are used to evaluate the information gain for the task in inspection step.

In this work, we propose a framework for sensor placement to maximize the information gain called Two-step Uncertainty network (TUN). The pipeline of TUN is shown in Figure 3.1 with red arrows. TUN consists of two steps, namely the imagination step and the inspection step. TUN firstly "imagines" the possible measurements at the un-observed locations. Then it estimates the task-specific information gain with the imagined measurements along with the previous obser-

vations in the inspection step. Both steps are deployed with the pre-trained neural networks. Given the task-specific information gain at all the un-observed locations, the agent adapts a greedy algorithm to select the optimal next location to inquire. This procedure emulates how we human think in such tasks: given the observations, we firstly imagine the possible outcomes at un-observed locations, then inspect the information pertaining to the task based on those possible outcomes. We will derive the proposed framework in the next section.

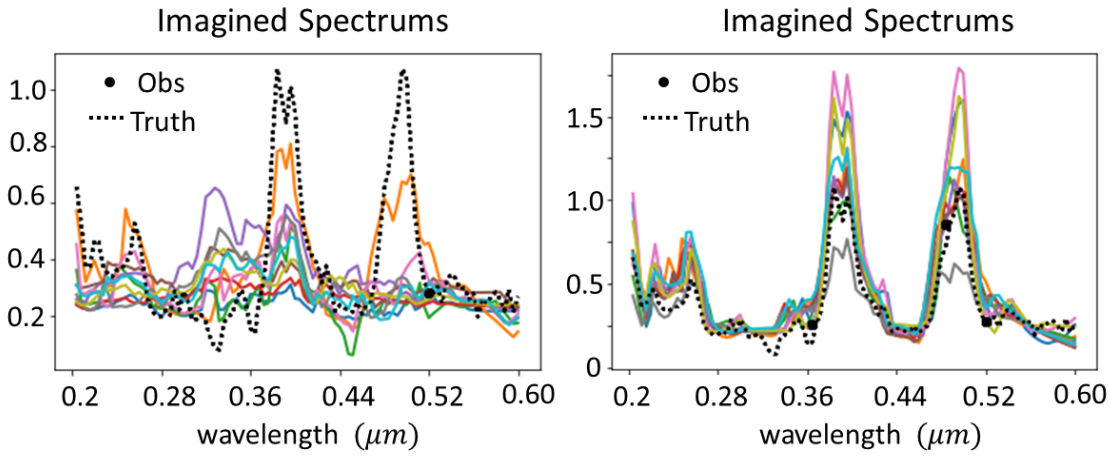

Figure 3.2: Example of the generation step on 1D spectrum dataset. The dashed line is the true spectrum and the solid spots are the observations. The generated instances are in colorful solid lines. Left: 10 imagined spectrums based on single observation. Right: 10 imagined spectrums based on three observations. The variation in the generated samples is mainly from scales and is independent of the task.

3.1.1 Objective Function in Adaptive Sensing

Consider a sequential sensing strategy, we denote locations as v and measurements as x. At the k^{th} step, we have the previous $k-1$ observations $Obs = \{x_1, v_1, ..., x_{k-1}, v_{k-1}\}$, the optimal location v_k^* is the one maximizes the mutual information (MI) between the object's label y and the possible

measurement x_k given the previous observations:

$$v_k^* = \arg\max_{v_k} MI(y; x_k | Obs, v_k) \tag{3.1}$$

The mutual information can be expressed as,

$$MI(y, x_k | v_k, Obs) = H(y|Obs) - \mathbb{E}_{Pr(x_k|v_k,Obs)} H(y|x_k, v_k, Obs) \tag{3.2}$$

where $\mathbb{E}$ is the expectation operator. In Equation 3.2, the first term is the uncertainty of labels conditioned on the previous observations. The second term is the expected uncertainty conditioned on observations and possible measurements x_k at v_k. The subtraction gives the uncertainty reduction or the information gain at the location v_k. It is worth noting that the first term is independent of v_k, and can be treated as a constant in optimizing Equation 3.2 with respect to v_k. The second term in Equation 3.2 can be approximated with Monte Carlo estimator as

$$MI(y, x_k | Obs) = -\sum_{m=1}^{M} \frac{1}{M} H(y|x_k^m, v_k, Obs) + Const. \tag{3.3}$$

where

$$x_k^m \sim Pr(x_k | v_k, Obs) \tag{3.4}$$

The summation in Equation 3.3 (without the negative sign) is the approximate remaining entropy with the measurement at v_k given. Maximizing the mutual information is equivalent to minimizing the remaining entropy. Equation 3.4 is the conditional distribution of the measurement at location v_k.

Following Equation 3.3-3.4, it is natural to approach the remaining entropy in two steps : (1), Generating instances of x_k that follows the distribution in Equation 3.4. (2), Evaluating the remaining entropy $\sum \frac{1}{M} H(y|x_k^m, v_k, Obs)$ with the generated instances in step (1). The graphical

models of the two steps are shown in Figure 3.1b. The design of our Two-step uncertainty network (TUN) follows the same rationale. In the imagination step, TUN generates multiple instances with a generative neural network. In the inspection step, a deterministic deep neural network is used to estimates the label distribution, and thus to evaluate the information gain(or negative remaining entropy).

3.1.2 Near Optimal Strategy:Two-step Uncertainty Network

The first step of TUN is to generate instances $x_k \sim Pr(x_k|v_k, Obs)$. Modeling the distribution of high dimensional data such as images is difficult and computationally expensive, even within a constrained family of distributions and simplified assumptions. Deterministic neural networks such as convolutions neural networks have shown strong expressiveness[48], but can not be used as generative models. Recently, remarkable progress has been made on modeling complex distribution of high dimensional data with generative neural network(GNN)[49, 50]. GNN maps the instances of re-parameterizable distribution (for example, multivariate normal distributions) to the instances of target complex distribution[51]. GNN is trained to maximize the log likelihood of the generated instances,

$$\log Pr(x_k|v_k, Obs) = \log \int Pr(x_k|z, v_k) Pr(z|Obs) dz \tag{3.5}$$

where z is the latent variable of multivariate normal distributions. The log likelihood with the integral is intractable. Thus, the evidence lower bond(ELBO) as an approximation is evaluated instead,

$$\ln \int Pr(x_k|z, v_k) Pr(z|Obs) dz \geq \mathbb{E}_{q_\phi(z)} \ln Pr(x_k|z, v_k) - D_{KL}[q_\phi(z); p_\theta(z)|x_k, v_k, Obs] \tag{3.6}$$

The posterior distribution $q_\phi(z|x_k, v_k, Obs)$ is conditioned on the previous $k-1$ observations Obs and the observed k^{th} measurement in the training data. The prior distribution $p_\theta(z|v_k, Obs)$ is

only conditioned on Obs. D_{KL} is the KL divergence measuring the difference between the two distributions. After training, we will have a generator network $G(v_k, Obs)$ that generates instances of measurement x_k. The generator network G is then employed in the imagination step of TUN.

The second step in TUN is inspection, which estimates the task-specific information gain(or negative remaining entropy) at v_k. With the generated M instances of x_k in the imagination step, the remaining entropy in Equation 3.3 can be expressed as

$$\sum_{m=1}^{M} \frac{1}{M} H(y|x_k, v_k, Obs) = -\sum_{m=1}^{M} \frac{1}{M} \sum_{y} Pr(y|x_k^m, v_k, Obs) \log Pr(y|x_k^m, v_k, Obs) \tag{3.7}$$

The remaining entropy is a function of the conditional distribution of the label $Pr(y|x_k^m, v_k, Obs)$. We approximate this conditional distribution with a deterministic neural network, the inspector network $D(x_k, v_k, Obs)$. Then the remaining entropy is evaluated as Equation 3.7. The most informative location to inquire is the one with the lowest remaining entropy. It can be shown that the true parameters of the model can be recovered by symptomatically maximizing a proper scoring rule[52]. A proper scoring rule S rewards the true distribution p more than any other distributions $\hat{p}$ on the training data d as

$$\int_d p(d) S(\hat{p}, d) \leq \int_d p(d) S(p, d) \tag{3.8}$$

It is shown that optimizing the softmax cross entropy loss function in the case of multi-class classification is equivalent to optimizing a proper scoring rule[53]. Thus, the inspector network D is trained with the softmax cross entropy loss. It is worth noting that the inspector network D takes different number of observations at different steps. To accept arbitrary number of observations, each observation is encoded separately with a shared-weight encoder, and the encoded vectors r_{enc} are aggregated to a fixed-length vector before the succeeding networks. To enforce the commutative property in the sequential sensing problem, we adapt a "mean operator" as the aggregator in TUN, which takes the average of the input vectors. We take random number of observations at the

randomly selected locations in the training stage.

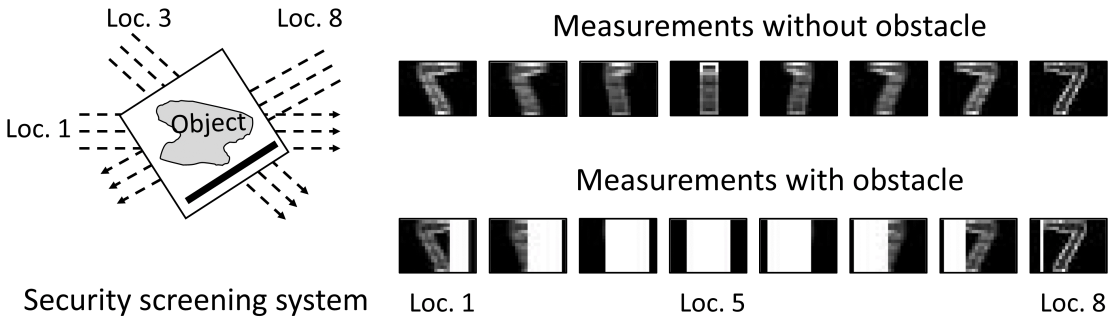

Figure 3.3: Left: x-ray security screening system. 8 different directions(locations) uniformly distributed within 180° are available to illuminate the object . The object is randomly rotated along z axis with an obstacle. Additive Gaussian noise is adapted on the measurements. Right: examples of measurements at 8 locations.

3.1.3 Results and Discussion

To evaluate the feasibility of TUN, we experimented with synthetic datasets. In the first experiment, we visualize the imagination step in TUN with simple 1D spectrum dataset. The spectrum dataset is generated from the spectrums of five minerals including Augite, Allanite, Xenotime, Bikitaitem and Pseudobrookite. We re-sampled the spectrums from $0.2\mu m$ to $0.6\mu m$ with 100 points and normalized them. The normalized spectrums are then scaled by a random factor ranging from 0.025 to 2.5 and corrupted by a zero-mean Gaussian noise with standard deviation of 0.03. The random scaling creates the intra-class uncertainty in the dataset. We prepared 5000 instances in the training dataset and 500 instances in the test dataset. The generator network G was trained to generate the instances of the spectrum with several observations given. The number and locations of the observations are randomly selected in the training process. In the test stage, we show 10 generated instances in colorful solid lines in Figure 3.2 with different observations. The observations are indicated as filled circle, and the true spectrum is shown in the dashed line. Given the single observation, the generated instances vary in both mineral types(inter-class uncertainty) and

scales(intra-class uncertainty). With three selected observations, the imagined spectrums mainly vary in scales. The intra-class uncertainty in the latter case is task-independent information indicating that the generator network G believes that not much information of label is remaining, given the observations. In other words, taking more measurements may not benefit the label prediction much. Thus, the agent may stop inquiring to avoid redundant inquiries. We will show more quantitative results of the evaluation of the task-specific information gain in the inspection step of TUN in high dimensional dataset.

To quantitatively evaluate the information gain(or uncertainty reduction), we created a high dimensional synthetic dataset from x-ray baggage scanning system in security screening. The physical model is shown in Figure 3.3. The objects to be screened are 3D digits with an obstacle that partially blocks the objects. The existence of the obstacle results in significant variation of information among different locations. There are 8 locations(angles) to illuminate the x-ray onto the object (ranging from 0° to 157.5°). Inquiry on each location returns a 2D projected image. The goal is to recognize the label of the object confidently with least number of inquiries. Before any inquiries, a randomized rotation is applied on the object along z axis. An additive Gaussian noise $n \sim (0, 0.02)$ is adopted on the observed images. TUN is trained on 5000 objects with random locations, effectively making the training dataset much larger. TUN is tested on 3000 set of observations generated from 1000 held-out un-seen objects. The generator network G firstly encodes the information of one or more observations into a fixed length representation vector. Then it generates M instances of possible measurements at the un-observed locations. based on the representation vector. In our model, $M = 10$. We show this process in Figure 3.4. In Figure 3.4 Left, the first measurement at location 4 is observed. We can see a corner feature in the measurement in the yellow box. Obviously the information from the observation is insufficient to reconstruct the 3D object, needless to mention the label of the object. We show three generated instances from the generator network at location 6 and 7 which are different in labels(digit 7 and 0) and fonts, yet consistent with the observation. In Figure 3.4 Middle, measurements at location 4 and 5 are

both obtained. The generated samples at location 6 and 7 are more convergent to the ground truth as more information in observations are extracted. The generator network in this situation almost collapses to a deterministic neural network. This shows that our generator network generates the samples following the distribution $x_k \sim Pr(x_k|v_k, Obs)$.

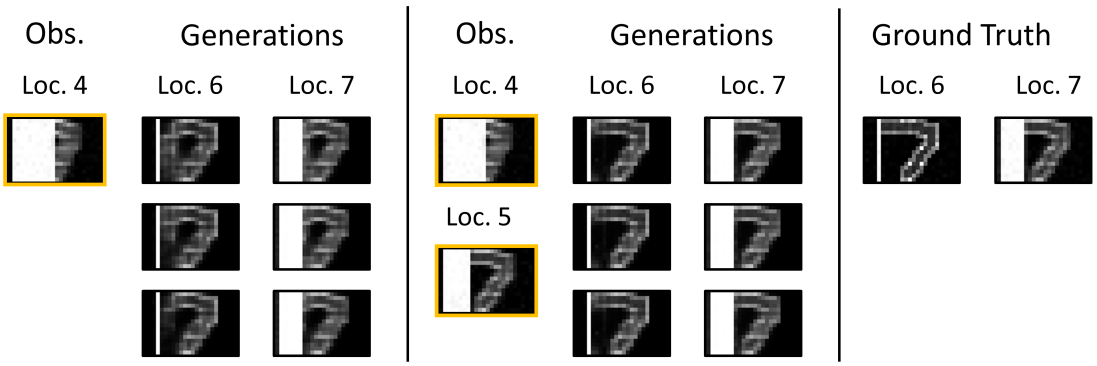

Figure 3.4: The imagination step of TUN on high dimensional dataset. Left: The 3 instances of the generations at location 6 and 7 given the observations at location 4. The generated instances are different in labels(digit 0 and 7) and fonts, but they all keep the corner feature occurred in the observation.Middle: The generations given two observations at location 4 and 5. The generated instances are much more convergent as more information is extracted from the observations. Right: The ground truth of the measurements at location 6 and 7.

In the second step of TUN, the generated samples are fed into the inspector network D, which estimates the probability of the labels $Pr(y|x_k, v_k, Obs)$ and evaluates the task specific information gain. We will perform both qualitative and quantitative analysis on the task specific information gain with TUN in the following paragraph. Firstly, we visualize the intermediate feature space in the initial sensing step of an example shown in Figure 3.5 Left. The obtained observations at location 5 is non-informative. We generated 100 instances at each location and fed them into the inspector network and visualized the feature space in the inspector network using t-SNE[54]. We select the vector at the layer before logits in the inspector network to visualize. The feature space is colored by locations and divided into three regions. The region 1 covers the features of the generated instances from exact observed location (location 5). Region 2 covers the

features from the locations close to the observed one (location 4 and 6). Region 3 contains the features from the locations far from the observed location. Clearly the features get more disperse as the distance to the observed location increases. This indicates that rich information lies in the locations in region 3. Although this is a qualitative analysis on the feature space, it justifies the necessity to sample multiple instances from the generator network. We will perform quantitative analysis of this example with the inspector network in next paragraph.

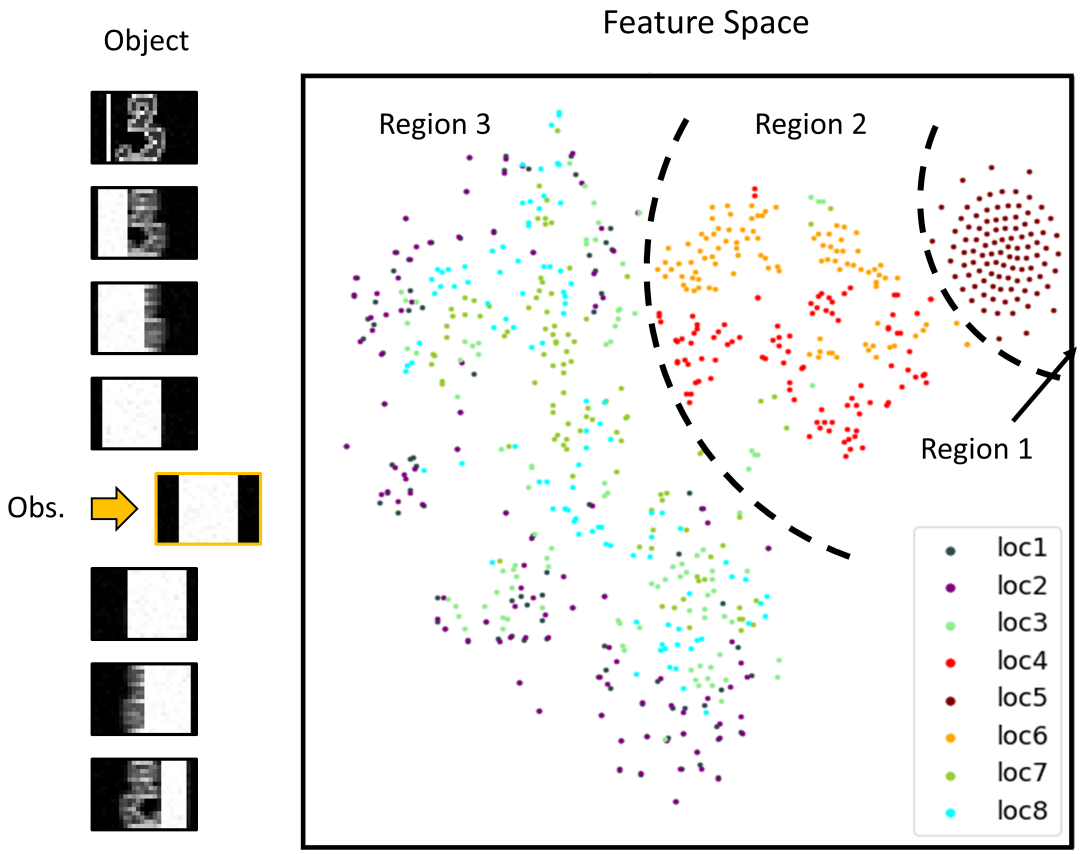

Figure 3.5: Task-specific feature distribution. We visualize the initial intermediate feature distribution in the inspector network using t-SNE. Clearly the features become more disperse at further locations from the observed location. The feature space is colored by locations. Region 1: Features of generated instances at the observed location. Region 2: Features at locations that are close to the observed location. Region 3: away from observed location. Region 3: Features at locations far from the observed locations.

The information gain is evaluated from the averaged entropy in Equation 3.7 from M samples. The averaged entropy indicates the estimated remaining uncertainty of the labels after we obtain the observation at location v_k. Our strategy is to pick the location with least remaining uncertainty, which equivalently maximizes the mutual information in Equation 3.2 (note the negative sign before the entropy). We show this quantitatively in the example in Figure 3.6. This is the same example as described in Figure 3.5. The first observation is at location 5 shown in yellow box in Figure 3.6a bottom, which is a non-informative observation. The averaged entropy for next step is shown in Figure 3.6a top.

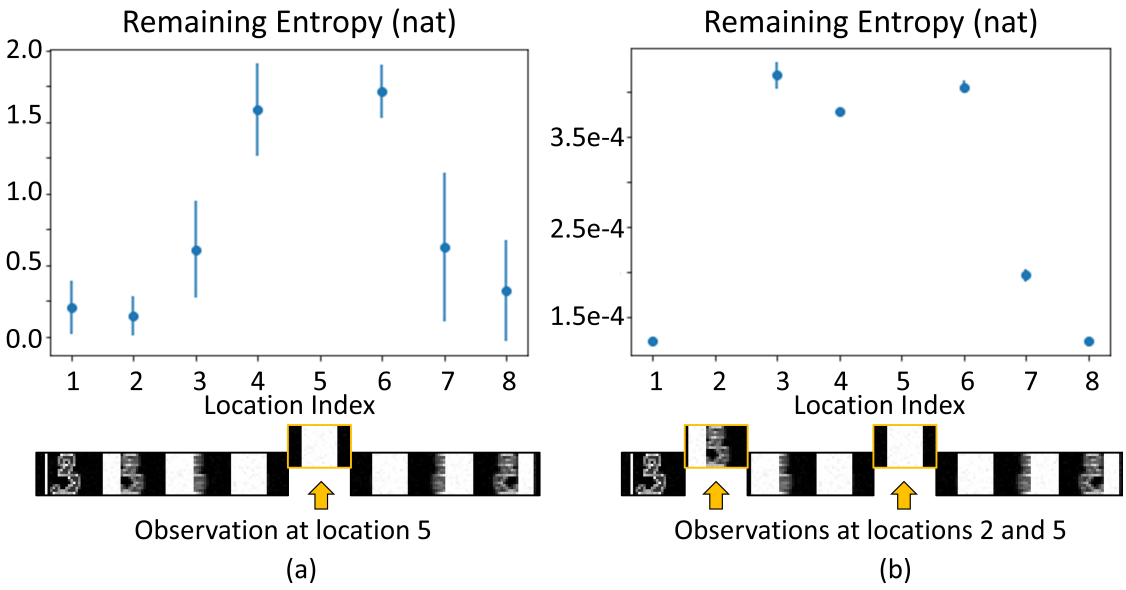

Figure 3.6: Entropy estimation: (a) The remaining entropy at un-observed locations given a single observation at location 5. The optimal next location in this step is location 2 at which the expected remaining entropy is least. (b) As we obtain the measurement at location 2 following TUN, the estimated remaining entropy at all the un-observed locations are almost zeros, indicating that the model is quite confident about the label with those observations.

The entropy plot estimates the potential remaining uncertainty at the un-observed locations, thus the less remaining uncertainty the better the location is. The entropy is averaged from 10 samples, and the standard deviation is shown as bars in Figure 3.6.

The entropy plot at the initial step indicates that our model believes there is less remaining uncertainty after we obtain the measurement at location 1,2, or 8 than that at location 3,4,6, or 7. Thus the next location selected by TUN is location 2 (which has least averaged remaining entropy). The successive estimation for the entropy is shown in Figure 3.6b in which the agent inquires and obtains the observation at location 2 following TUN. With the observations at location 2 and 5, the remaining entropy is very low with neglectable variance. This entropy plot shows TUN is quite confident on the label of the object, and believes there is not much information left at un-observed locations. A threshold can be used as a stopping criterion in practice. We compare TUN with random sampling strategy(RS) and Gaussian Process(GP) strategy. We adapt squared exponential kernel in GP and employ the 2D coordinates and the projection angle $[x, y, cos\theta, sin\theta]$ as features. The GP model is fitted with training data and performs the prediction of measurement at un-observed locations. To evaluate the strategies, we trained classifiers with different number of the observations. The training data for the classifiers are generated from 5000 held-out objects with random noise and rotation. All the observations in training the classifiers are taken from random sampling strategy. The performance in both accuracy and entropy(confidence) with different sensing budget are shown in Figure 3.7. The first location is randomly selected in all three strategies, thus, the performances are the same for all strategies. Start from the second step, TUN outperforms other strategies consistently with higher accuracy and less uncertainty.

3.2 Limitation and Outlook

In this work, we present a task-driven sensor placement framework to maximize the information gain. The proposed framework (TUN) is able to perceive and understand the observation, approximate the conditional distribution of the object, and estimate the information gain pertaining to the task. In the security screening experiment we demonstrated, TUN outperforms random strategy and GP strategy consistently. The limitation of the method is the requirement of sufficient data

and computing resource to train the neural networks. We trained the networks in our two NVIDIA GPUs(RTX 2080 ti) for 36 hours. On the other hand, it takes seconds to run the trained model in test time, which has the potential in real-time applications.

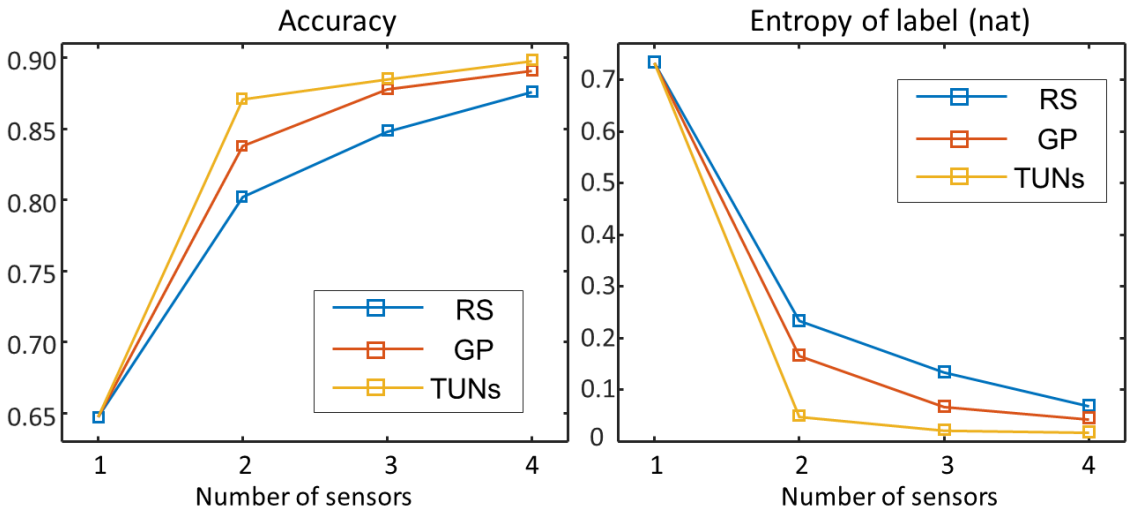

Figure 3.7: Accuracy and entropy at different numbers of inquiries for TUN, Gaussian process(GP) based and random sampling(RS) strategies. The advantages of sensor placement strategies starts at two observations. The proposed framework, TUN, outperforms GP and RS strategies consistently.

CHAPTER 4: CONCLUSION

4.1 Summary of Contributions

In conclusion, we have presented 1) the methods to improve temporal resolution in active and passive range imaging; 2) the method to estimate the object distribution as well as the task-specific information gain in adaptive sensing(sensor placement) tasks. These methods employ modified optimization frameworks and neural networks to surpass the limits of the conventional computational imaging methods. We summarize our contributions below.

To passively image the high-speed 3D scenes, we propose a novel stereo imaging system in which the projected high-speed frames on one of the perspectives are modulated by DMD. We also propose a novel two-step framework to reconstruct the high-speed 3D scenes. In the first step, the projected high-speed frames from the modulated perspective are reconstructed by TwIST algorithm. In the second step, the depth of each pixel in the reconstructed high-speed images are estimated with modified graph cut method. We show the reconstructed 3D frames of a moving baseball at 800 fps with the cameras operating at 80 fps.

To actively image the high-speed 3D scenes, we propose a novel structured light imaging system in which the illumination is high-speed pseudo-random binary patterns. Different from the conventional structured light systems, our system is capable of recording the temporal information at the frequency higher than the sampling rate of the camera, and the depth information of the scene. We propose a novel alternating optimization framework to reconstruct the depth of the scene with the carefully calibrated system. We show the reconstructed 1000 fps 3D frames with the camera operating at 200 fps.

To maximize the information gain in each step of sensor placement tasks, we propose a novel two-step framework. In the first step, we build generative neural networks to capture the distribution of objects conditioned on the observations. In the second step, we build the deep neural

networks to extract the task-specific information gain from the generated objects. The experiments on the synthetic dataset of security screening task show that our proposed framework outperforms random strategy and Gaussian process based strategy in both accuracy and confidence(entropy).

4.2 Future Outlook

We have presented a set of frameworks to overcome the limitations of conventional computational imaging methods. Future work may include applying the frameworks to other practical applications. For instance, our range imaging systems in Chapter 2 shows 5-10 folds improvement on temporal resolution and the capability of 3D imaging. A possible direction is to apply our modulation path and reconstruction framework into camera array systems for high-speed imaging which inherently records multiple perspectives of the scene. The adaptive sensing framework in Chapter 3 is capable of selecting the most informative location to sense by modeling the distribution of objects from large amount of data. One possible direction is to apply this framework in brain CT scanning which is very dose sensitive and is not expensive to obtain sufficient labeled data.

LIST OF REFERENCES

1. Michael R Peres. *The focal encyclopedia of photography*. Taylor & Francis, 2013.
2. John Hannavy. *Encyclopedia of nineteenth-century photography*. Routledge, 2013.
3. Michael R Peres. *The concise Focal encyclopedia of photography: from the first photo on paper to the digital revolution*. CRC Press, 2014.
4. Yangyang Sun and Shuo Pang. “Multi-perspective scanning microscope based on Talbot effect”. In: *Applied Physics Letters* 108.2 (2016), p. 021102.
5. Yangyang Sun and Shuo Pang. “Fluorescence Talbot microscope using incoherent source”. In: *Journal of biomedical optics* 21.8 (2016), p. 086003.
6. Jialei Tang, Yangyang Sun, Shuo Pang, and Kyu Young Han. “Spatially encoded fast single-molecule fluorescence spectroscopy with full field-of-view”. In: *Scientific reports* 7.1 (2017), p. 10945.
7. P Mansfield and PK Grannell. “" Diffraction" and microscopy in solids and liquids by NMR”. In: *Physical Review B* 12.9 (1975), p. 3618.
8. Patrick Llull, Xuejun Liao, Xin Yuan, Jianbo Yang, David Kittle, Lawrence Carin, Guillermo Sapiro, and David J Brady. “Coded aperture compressive temporal imaging”. In: *Optics express* 21.9 (2013), pp. 10526–10545.
9. Leonid I Rudin, Stanley Osher, and Emad Fatemi. “Nonlinear total variation based noise removal algorithms”. In: *Physica D: nonlinear phenomena* 60.1-4 (1992), pp. 259–268.
10. Emmanuel J Candes, Justin K Romberg, and Terence Tao. “Stable signal recovery from incomplete and inaccurate measurements”. In: *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences* 59.8 (2006), pp. 1207–1223.

11. Michael Elad and Michal Aharon. “Image denoising via sparse and redundant representations over learned dictionaries”. In: *IEEE Transactions on Image processing* 15.12 (2006), pp. 3736–3745.

12. Yangyang Sun, Xin Yuan, and Shuo Pang. “Compressive high-speed stereo imaging”. In: *Optics express* 25.15 (2017), pp. 18182–18190.

13. José M Bioucas-Dias and Mário AT Figueiredo. “A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration”. In: *IEEE Transactions on Image processing* 16.12 (2007), pp. 2992–3004.

14. Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. “Distributed optimization and statistical learning via the alternating direction method of multipliers”. In: *Foundations and Trends® in Machine learning* 3.1 (2011), pp. 1–122.

15. David R Hunter and Kenneth Lange. “A tutorial on MM algorithms”. In: *The American Statistician* 58.1 (2004), pp. 30–37.

16. Neal Parikh, Stephen Boyd, et al. “Proximal algorithms”. In: *Foundations and Trends® in Optimization* 1.3 (2014), pp. 127–239.

17. Amir Beck and Marc Teboulle. “Gradient-based algorithms with applications to signal recovery”. In: *Convex optimization in signal processing and communications* (2009), pp. 42–88.

18. Yangyang Sun, Xin Yuan, and Shuo Pang. “High-speed compressive range imaging based on active illumination”. In: *Optics express* 24.20 (2016), pp. 22836–22846.

19. Xin Yuan, Yangyang Sun, and Shuo Pang. “Compressive video sensing with side information”. In: *Applied optics* 56.10 (2017), pp. 2697–2704.

20. Xin Yuan and Shuo Pang. “Compressive video microscope via structured illumination”. In: *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2016, pp. 1589–1593.

21. Yangyang Sun, Zheyuan Zhu, and Shuo Pang. “Learning models for acquisition planning of CT projections (Conference Presentation)”. In: *Anomaly Detection and Imaging with X-Rays (ADIX) IV*. Vol. 10999. International Society for Optics and Photonics. 2019, p. 109990C.

22. Fabian Rengier, Amit Mehndiratta, Hendrik Von Tengg-Kobligk, Christian M Zechmann, Roland Unterhinninghofen, H-U Kauczor, and Frederik L Giesel. “3D printing based on imaging data: review of medical applications”. In: *International journal of computer assisted radiology and surgery* 5.4 (2010), pp. 335–341.

23. Philip Treleaven and Jonathan Wells. “3D body scanning and healthcare applications”. In: *Computer* 40.7 (2007), pp. 28–34.

24. Anthony Corns and Robert Shaw. “High resolution 3-dimensional documentation of archaeological monuments & landscapes using airborne LiDAR”. In: *Journal of Cultural Heritage* 10 (2009), e72–e77.

25. Wijerupage Sardha Wijesoma, KR Sarath Kodagoda, and Arjuna P Balasuriya. “Road-boundary detection and tracking using ladar sensing”. In: *IEEE Transactions on robotics and automation* 20.3 (2004), pp. 456–464.

26. Martial Hebert. “Active and passive range sensing for robotics”. In: *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*. Vol. 1. IEEE. 2000, pp. 102–110.

27. Leonid I Rudin and Stanley Osher. “Total variation based image restoration with free local constraints”. In: *Proceedings of 1st International Conference on Image Processing*. Vol. 1. IEEE. 1994, pp. 31–35.

28. Heiko Hirschmuller. “Accurate and efficient stereo processing by semi-global matching and mutual information”. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*. Vol. 2. IEEE. 2005, pp. 807–814.

29. Vladimir Kolmogorov and Ramin Zabih. *Computing visual correspondence with occlusions via graph cuts*. Tech. rep. Cornell University, 2001.

30. Zhengyou Zhang. “A flexible new technique for camera calibration”. In: *IEEE Transactions on pattern analysis and machine intelligence* 22 (2000).

31. Yuri Boykov and Vladimir Kolmogorov. “An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision”. In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 9 (2004), pp. 1124–1137.

32. Vladimir Kolmogorov and Ramin Zabih. “What energy functions can be minimizedvia graph cuts?” In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 2 (2004), pp. 147–159.

33. Patrick Llull, Xin Yuan, Lawrence Carin, and David J Brady. “Image translation for single-shot focal tomography”. In: *Optica* 2.9 (2015), pp. 822–825.

34. Xin Yuan. “Generalized alternating projection based total variation minimization for compressive sensing”. In: *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2016, pp. 2539–2543.

35. Xuejun Liao, Hui Li, and Lawrence Carin. “Generalized Alternating Projection for Weighted-2,1 Minimization with Applications to Model-Based Compressive Sensing”. In: *SIAM Journal on Imaging Sciences* 7.2 (2014), pp. 797–823.

36. Fabrice Gamboa, Elisabeth Gassiat, et al. “Bayesian methods and maximum entropy for ill-posed inverse problems”. In: *The Annals of Statistics* 25.1 (1997), pp. 328–350.

37. Chengyu Hu, Ming Li, Deze Zeng, and Song Guo. “A survey on sensor placement for contamination detection in water distribution systems”. In: *Wireless Networks* 24.2 (2018), pp. 647–661.

38. Linh V Nguyen, Sarath Kodagoda, Ravindra Ranasinghe, and Gamini Dissanayake. “Information-driven adaptive sampling strategy for mobile robotic wireless sensor network”. In: *IEEE Transactions on Control Systems Technology* 24.1 (2015), pp. 372–379.

39. Wieslaw Ostachowicz, Rohan Soman, and Pawel Malinowski. “Optimization of sensor placement for structural health monitoring: A review”. In: *Structural Health Monitoring* 18.3 (2019), pp. 963–988.

40. Ahmad Masoudi, Ratchaneekorn Thamvichai, and Mark A Neifeld. “Shape threat detection via adaptive computed tomography”. In: *Anomaly Detection and Imaging with X-Rays (ADIX)*. Vol. 9847. International Society for Optics and Photonics. 2016, 98470H.

41. S Ouadah, M Jacobson, Joseph Webster Stayman, T Ehtiati, Clifford Weiss, and JH Siewerdsen. “Task-driven orbit design and implementation on a robotic C-arm system for cone-beam CT”. In: *Medical Imaging 2017: Physics of Medical Imaging*. Vol. 10132. International Society for Optics and Photonics. 2017, 101320H.

42. Michael R Garey and David S Johnson. *Computers and intractability*. Vol. 29. wh freeman New York, 2002.

43. George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. “An analysis of approximations for maximizing submodular set functions—I”. In: *Mathematical programming* 14.1 (1978), pp. 265–294.

44. Andreas Krause, Ajit Singh, and Carlos Guestrin. “Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies”. In: *Journal of Machine Learning Research* 9.Feb (2008), pp. 235–284.

45. Carl Edward Rasmussen. “Gaussian processes in machine learning”. In: *Summer School on Machine Learning*. Springer. 2003, pp. 63–71.

46. David Ha and Jürgen Schmidhuber. “Recurrent world models facilitate policy evolution”. In: *Advances in Neural Information Processing Systems*. 2018, pp. 2450–2462.

47. Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. “Deep reinforcement learning that matters”. In: *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

48. Jian Zhao, Yangyang Sun, Zheyuan Zhu, Jose Enrique Antonio-Lopez, Rodrigo Amezcua Correa, Shuo Pang, and Axel Schulzgen. “Deep learning imaging through fully-flexible glass-air disordered fiber”. In: *ACS Photonics* 5.10 (2018), pp. 3930–3935.

49. Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. “Draw: A recurrent neural network for image generation”. In: *arXiv preprint arXiv:1502.04623* (2015).

50. SM Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S Morcos, Marta Garnelo, Avraham Ruderman, Andrei A Rusu, Ivo Danihelka, Karol Gregor, et al. “Neural scene representation and rendering”. In: *Science* 360.6394 (2018), pp. 1204–1210.

51. Diederik P Kingma and Max Welling. “Auto-encoding variational bayes”. In: *arXiv preprint arXiv:1312.6114* (2013).

52. Tilmann Gneiting and Adrian E Raftery. “Strictly proper scoring rules, prediction, and estimation”. In: *Journal of the American Statistical Association* 102.477 (2007), pp. 359–378.

53. Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. “Simple and scalable predictive uncertainty estimation using deep ensembles”. In: *Advances in Neural Information Processing Systems*. 2017, pp. 6402–6413.

54. Laurens van der Maaten and Geoffrey Hinton. “Visualizing data using t-SNE”. In: *Journal of machine learning research* 9.Nov (2008), pp. 2579–2605.