Depth Reconstruction from Sparse Samples: Representation, Algorithm, and Sampling

Lee-Kang Liu, *Student Member, IEEE*, Stanley H. Chan, *Member, IEEE* and Truong Q. Nguyen, *Fellow, IEEE*

Abstract—The rapid developments of 3D technology and computer vision applications have motivated a thrust of methodologies for depth acquisition and estimation. However, existing hardware and software acquisition methods have limited performance due to poor depth precision, low resolution and high computational cost. In this paper, we present a computationally efficient method to estimate dense depth maps from sparse measurements. There are three main contributions. First, we provide empirical evidence that depth maps can be encoded much more sparsely than natural images by using common dictionaries such as wavelets and contourlets. We also show that a combined wavelet-contourlet dictionary achieves better performance than using either dictionary alone. Second, we propose an alternating direction method of multipliers (ADMM) for depth map reconstruction. A multi-scale warm start procedure is proposed to speed up the convergence. Third, we propose a two-stage randomized sampling scheme to optimally choose the sampling locations, thus maximizing the reconstruction performance for a given sampling budget. Experimental results show that the proposed method produces high quality dense depth estimates, and is robust to noisy measurements. Applications to real data in stereo matching are demonstrated.

Index Terms—Sparse reconstruction, random sampling, wavelet, contourlet, disparity estimation, alternating direction method of multipliers, compressed sensing

I. Introduction

The rapid development of 3D technology has created a new wave of visualization and sensing impacts to the digital signal processing community. From remote sensing [1] to preserving historical heritages [2], and from rescue [3] to 3D laparoscopic surgery [4], [5], the footprints of 3D have been influencing a broad spectrum of the technological frontiers.

The successful development of 3D signal processing is fundamentally linked to a system's ability to acquire depth. To date, there are two major classes of depth acquisition techniques: hardware solutions and computational procedures.

L. Liu and T. Nguyen are with Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA 92093, USA. Emails: l7liu@ucsd.edu and tqn001@eng.ucsd.edu

S. Chan is with School of Electrical and Computer Engineering and Department of Statistics, Purdue University, West Lafayette, IN 47907, USA. Email: stanleychan@purdue.edu

This paper follows the concept of reproducible research. All the results and examples presented in the paper are reproducible using the code and images available online at http://videoprocessing.ucsd.edu/.

Hardware devices are usually equipped with active sensors such as time-of-flight (ToF) camera [6] and LiDAR [7]. While being able to produce high quality depth maps, these hardware systems have high instrumentation cost. Moreover, the data acquisition time of the devices is long (*e.g.*, a recently proposed ToF cameras can only achieve 10fps [8], whereas standard cameras nowadays can easily achieve 60fps.) Although speeding up is possible, spatial resolution has to be traded off in return.

An alternative solution to acquiring depth is to estimate depth using a set of computational procedures. This class of computational methods, broadly referred to as disparity estimation algorithms [9–12], estimates the depth by computing the disparities between a pair of stereo images through their corresponding matching points [13], [14]. Disparity estimation algorithms usually work well under well conditioned environments, but they could be sensitive to illumination, noise, stereo alignments, and other camera factors. Thus, the effective number of matching points that one can use for disparity estimation is actually much fewer than the number of pixels of the depth map [15], [16].

A. Scope and Contributions

The objective of this paper is to present a sampling and reconstruction framework to improve and speed up the depth acquisition process. The key idea is to carefully select a sparse subset of spatial samples and use an optimization algorithm to reconstruct the final dense depth map.

The three major contributions of this paper are as follows.

1) Representation (Section III). In order to reconstruct the depth map, we must first define an appropriate representation. We show that, as opposed to natural images, depth maps can be well approximated using a sparse subset of wavelet bases. Moreover, we show that a combined dictionary of wavelets and contourlets can further improve the reconstruction quality.

2) Algorithm (Section IV). We propose a fast numerical algorithm based on the alternating direction method of multipliers (ADMM). We derive novel splitting strategies that allow one to solve a sequence of parallelizable subproblems. We also present a multiscale implementation that utilizes the depth structures for efficient warm starts.

3) Sampling (Section V). We propose an efficient spatial sampling strategy that maximizes the reconstruction performance. In particular, we show that for a fixed sampling budget, a high quality sampling pattern can be obtained by allocating random samples with probabilities in proportional to the magnitudes of the depth gradients.

B. Related Work

The focus of this paper lies in the intersection of two closely related subjects: depth enhancement and compressed sensing. Both subjects have a rich collection of prior works but there are also limitations which we should now discuss.

The goal of depth enhancement is to improve the resolution of a depth map. Some classical examples include Markov Random Field (MRF) [17], bilateral filter [18], and other approaches [19], [20]. One limitation of these methods is that the low-resolution depth maps are sampled uniformly. Also, it is usually assumed that a color image of the scene is available. In contrast, our proposed method is applicable to *any* non-uniformly sampled low-resolution depth map and does not require color images. Thus, the new method allows for a greater flexibility for the enhancement.

Compressed sensing (CS) is a popular mathematical framework for sampling and recovery [21]. In many cases, CS methods assume that *natural images* exhibit sparse structures in certain domains, *e.g.*, wavelet. However, as will be discussed in Section III of this paper, natural images are indeed *not* sparse. If we compare natural images to depth maps, the latter would show a much sparser structure than the former. Furthermore, the theory of combined bases [22], [23] shows that a pair of incoherent bases are typically more effective for signal recovery. Yet, the application of these theories to depth maps is not fully explored.

The most relevant paper to our work is perhaps [24]. However, our work has two advantages. First, we propose a new ADMM algorithm for the reconstruction task (Section IV). We show that the ADMM algorithm is significantly more efficient than the subgradient method proposed in [24]. Second, we present a sampling scheme to choose optimal sampling patterns to improve the depth reconstruction (Section V), which was not discussed in [24].

We should also mention a saliency-guided CS method proposed in [25], [26]. In these two papers, the spatial sampling is achieved by a mixing-plus-sampling process, meaning that the unknown pixels are filtered and then sub-sampled. The filtering coefficients are constructed through a pre-defined saliency map and certain density functions (*e.g.*, Gaussian-Bernoulli). In our work, the mixing process is *not* required so that depth values are sampled without filtering. This makes our proposed method applicable to disparity estimation where mixing cannot be used (otherwise it will defeat the purpose of reconstructing dense depth maps from a few estimated values.)

Finally, advanced computational photography techniques are recently proposed for fast depth acquisition, *e.g.*, [27], [28]. However, the problem settings of these works involve hardware designs and are thus different from this paper.

The rest of the paper is organized as follows. After elaborating the problem and clarifying notations in Section II, we discuss the representation of depth maps in Section III. A fast reconstruction algorithm is presented in Section IV. The design of optimal sampling patterns is discussed in Section V. Experimental results are shown in Section VI, and a concluding remark is given in Section VII.

II. NOTATIONS AND PROBLEM FORMULATION

In this section we introduce notations and elaborate on the problem formulation.

A. Depth and Disparity

The type of data that we are interested in studying is the depth map. Depth can be directly measured using active sensors, or inferred from the disparity of a pair of stereo images. Since the correspondence between depth and disparity is unique by simple geometry [29], in the rest of the paper we shall use depth and disparity interchangeably.

B. Sampling Model

Let $\boldsymbol{x} \in \mathbb{R}^N$ be an $N \times 1$ vector representing a disparity map. For simplicity we assume that $\boldsymbol{x}$ is normalized so that $0 \le x_j \le 1$ for $j = 1, \ldots, N$.

To acquire a set of spatial samples, we define a diagonal matrix $\boldsymbol{S} \in \mathbb{R}^{N \times N}$ with the (j, j)th entry being

$$S_{jj} \stackrel{\text{def}}{=} \begin{cases} 1, & \text{with probability } p_j, \\ 0, & \text{with probability } 1 - p_j, \end{cases} \quad (1)$$

where $\{p_j\}_{j=1}^N$ is a sequence of pre-defined probabilities. Specific examples of $\{p_j\}_{j=1}^N$ will be discussed below. For now, we only require $\{p_j\}_{j=1}^N$ to satisfy two criteria: (1) for each $j = 1, \ldots, N$, p_j must be bounded so that $0 \le p_j \le 1$; (2) the average of the probabilities must achieve a target *sampling ratio* ξ:

$$\frac{1}{N} \sum_{j=1}^N p_j = \xi, \quad (2)$$

where $0 < \xi < 1$.

Example 1: If $p_j = \xi$ for all j, then the sampling pattern $\boldsymbol{S}$ is a diagonal matrix with uniformly random entries. This sampling pattern corresponds to a uniform sampling without filtering in the classical compressed sensing, e.g., [21].

Example 2: If $p_j = 1$ for $j \in \Omega_1$ and $p_j = 0$ for $j \in \Omega_0$, where Ω_1 and Ω_0 are two pre-defined sets such that $|\Omega_1| = \xi N$ and $|\Omega_0| = (1 - \xi)N$, then $\boldsymbol{S}$ is a deterministic sampling pattern. In particular, if Ω_1 and Ω_0 are designed so that the indices are uniformly gridded, then $\boldsymbol{S}$ will become the usual down-sampling operator.

With $\boldsymbol{S}$, we define the sampled disparity map as

$$\boldsymbol{b} = \boldsymbol{S}\boldsymbol{x}. \quad (3)$$

Note that in (3), the sampled disparity $\boldsymbol{b} \in \mathbb{R}^{N \times 1}$ will contain zeros, *i.e.*, $b_j = 0$ if $S_{jj} = 0$. Physically, this corresponds to the situation where the unsampled pixels are marked with a value of zero.

Remark 1: Since $\boldsymbol{S}$ is a random diagonal matrix, readers at this point may have concerns about the overall number of samples which is also random. However, we argue that such randomness has negligible effects for the following reason. For large N, standard concentration inequality guarantees that the

average number of ones in S stays closely to ξN. In particular, by Bernstein's inequality [30] we can show that for $\varepsilon > 0$,

$$\Pr\left(\left|\frac{1}{N}\sum_{j=1}^{N} S_{jj} - \xi\right| > \varepsilon\right) \le 2\exp\left\{-\frac{N\varepsilon^2}{1/2 + 2\varepsilon/3}\right\}. \quad (4)$$

Therefore, although the sampling pattern in our framework is randomized, the average number of samples is concentrated around ξN for large N.

C. Representation Model

To properly formulate the reconstruction problem, we assume that the disparity map can be efficiently represented as a linear combination of basis vectors $\{\boldsymbol{\varphi}_i\}_{i=1}^{M}$:

$$\boldsymbol{x} = \sum_{i=1}^{M} \langle \boldsymbol{x}, \boldsymbol{\varphi}_i \rangle \boldsymbol{\varphi}_i, \quad (5)$$

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product. Defining $\alpha_i \overset{\text{def}}{=} \langle \boldsymbol{x}, \boldsymbol{\varphi}_i \rangle$ as the ith basis coefficient, $\boldsymbol{\alpha} \overset{\text{def}}{=} [\alpha_1, \ldots, \alpha_M]^T$, and $\boldsymbol{\Phi} \overset{\text{def}}{=} [\boldsymbol{\varphi}_1, \ldots, \boldsymbol{\varphi}_M]$, the relationship in (5) can be equivalently written as $\boldsymbol{x} = \boldsymbol{\Phi}\boldsymbol{\alpha}$.

The reconstruction problem can be posed as an optimization problem in which the goal is to seek a sparse vector $\boldsymbol{\alpha} \in \mathbb{R}^M$ such that the observed samples $\boldsymbol{b}$ are best approximated. Mathematically, we consider the problem

$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{S}\boldsymbol{\Phi}\boldsymbol{\alpha} - \boldsymbol{b}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1, \quad (6)$$

where $\lambda > 0$ is a regularization parameter, and $\|\cdot\|_1$ is the ℓ_1-norm of a vector.

In this paper, we are mainly interested in two types of $\boldsymbol{\Phi}$ — the wavelet frame and the contourlet frame [31]. Frames are generalizations of the standard bases in which M, the number of bases, can be more than N, the dimension of $\boldsymbol{x}$. Moreover, for any frame $\boldsymbol{\Phi}$, it holds that $\boldsymbol{\Phi}\boldsymbol{\Phi}^T = \boldsymbol{I}$. Therefore, $\boldsymbol{x} = \boldsymbol{\Phi}\boldsymbol{\alpha}$ if and only if $\boldsymbol{\alpha} = \boldsymbol{\Phi}^T \boldsymbol{x}$. Using this result, we can equivalently express (6) as

$$\underset{\boldsymbol{x}}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{S}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \lambda\|\boldsymbol{\Phi}^T \boldsymbol{x}\|_1. \quad (7)$$

Remark 2: In compressed sensing literature, (6) is known as the synthesis problem and (7) is known as the analysis problem [32]. Furthermore, the overall measurement matrix $\boldsymbol{S}\boldsymbol{\Phi}$ in (6) suggests that if $p_j = \xi$ for all j, then $\boldsymbol{S}\boldsymbol{\Phi}$ corresponds to the partial orthogonal system as discussed in [33]. In this case, the restricted isometry property (RIP) holds [34] and exact recovery can be guaranteed under appropriate assumptions of sparsity and number of measurements. For general $\{p_j\}_{j=1}^{N}$, establishing RIP is more challenging, but empirically we observe that the optimization produces reasonable solutions.

D. Penalty Functions

As discussed in [24], (7) is not an effective formulation because the ℓ_1 norm penalizes *both* the approximation (lowpass) and the detailed (highpass) coefficients. In reality, since disparity maps are mostly piecewise linear functions, the lowpass coefficients should be maintained whereas the highpass coefficients are desirable to be sparse. To this end, we introduce a binary diagonal matrix $\boldsymbol{W} \in \mathbb{R}^{M \times M}$ where the (j, j)th entry is 0 if j is an index in the lowest passband, and is 1 otherwise. Consequently, we modify the optimization problem as

$$\underset{\boldsymbol{x}}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{S}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \lambda\|\boldsymbol{W}\boldsymbol{\Phi}^T \boldsymbol{x}\|_1. \quad (8)$$

Finally, it is desirable to further enforce smoothness of the reconstructed disparity map. Therefore, we introduce a total variation penalty so that the problem becomes

$$\underset{\boldsymbol{x}}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{S}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \lambda\|\boldsymbol{W}\boldsymbol{\Phi}^T \boldsymbol{x}\|_1 + \beta\|\boldsymbol{x}\|_{TV}. \quad (9)$$

Here, the total variation norm is defined as

$$\|\boldsymbol{x}\|_{TV} \overset{\text{def}}{=} \|\boldsymbol{D}_x \boldsymbol{x}\|_1 + \|\boldsymbol{D}_y \boldsymbol{x}\|_1, \quad (10)$$

where $\boldsymbol{D} = [\boldsymbol{D}_x; \boldsymbol{D}_y]$ is the first-order finite difference operator in the horizontal and vertical directions. The above definition of total variation is known as the anisotropic total variation. The same formulation holds for isotropic total variation, in which $\|\boldsymbol{x}\|_{TV} = \sum_{j=1}^{N} \sqrt{[\boldsymbol{D}_x \boldsymbol{x}]_j^2 + [\boldsymbol{D}_y \boldsymbol{x}]_j^2}$.

The problem in (9) is generalizable to take into account of a combination of L dictionaries. In this case, one can consider a sum of L penalty terms as

$$\underset{\boldsymbol{x}}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{S}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \sum_{\ell=1}^{L} \lambda_\ell\|\boldsymbol{W}_\ell \boldsymbol{\Phi}_\ell^T \boldsymbol{x}\|_1 + \beta\|\boldsymbol{x}\|_{TV}. \quad (11)$$

For example, in the case of combined wavelet and contourlet dictionaries, we let $L = 2$.

III. Sparse Representation of Disparity Map

The choice of the dictionary $\boldsymbol{\Phi}$ in (11) is an important factor for the reconstruction performance. In this section we discuss the general representation problem of disparity maps. We show that disparity maps can be represented more sparsely than natural images. We also show that a combined wavelet-contourlet dictionary is more effective in representing disparity maps than using the wavelet dictionary alone.

A. Natural Images vs Depth Data

Seeking effective representations for *natural images* is a well-studied subject in image processing [31], [35–41]. However, representations of *disparity maps* seems to be less studied. For example, it is unclear how sparse can a predefined dictionary (*e.g.*, wavelets) encode disparity maps as compared to natural images.

To address this question, we consider a 128×128 cropped patch from a gray-scaled image and the corresponding patch in the disparity map. For each of the image and the disparity, we apply the wavelet transform with Daubechies $5/3$ filter and 5 decomposition levels. Then, we truncate the wavelet coefficients to the leading 5% coefficients with the largest magnitudes. The reconstructed patches are compared and the results are shown in Figure 1.

(a) Original disparity

(b) Approx. disparity (50.25 dB)
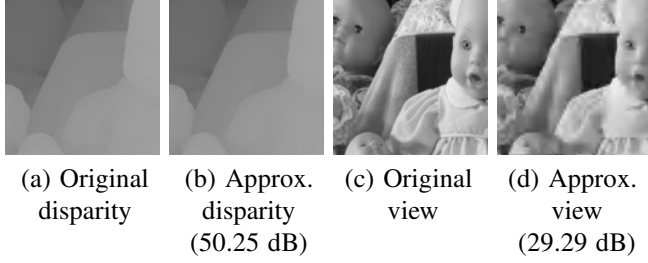
(c) Original view

(d) Approx. view (29.29 dB)

Fig. 1: PSNR values of approximating a disparity patch and a image patch using the leading 5% of the wavelet coefficients.

The result indicates that for the same number of wavelet coefficients, the disparity map can be synthesized with significantly lower approximation error than the image. While such result is not surprising, the big difference in the PSNRs provides evidence that reconstruction of disparity maps from sparse samples should achieve better results than that of natural images.

B. Wavelet vs Contourlet

The above results indicate that wavelets are efficient representations for disparity maps. Our next question is to ask whether some of the dictionaries would do better than other dictionaries.

1) Evaluation Metric: To compare the performance of two dictionaries, it is necessary to first specify what metric to use. For the purpose of reconstruction, we compare the mean squared error (MSE) of the reconstructed disparity maps obtained by feeding different dictionaries into (11). For any fixed sampling pattern S, we say that a dictionary Φ_1 is better than another dictionary Φ_2 if the reconstruction result using Φ_1 has a lower MSE than using Φ_2, for the best choice of parameters λ_1, λ_2 and β. Note that in this evaluation we do not compare the sparsity of the signal using different dictionaries. In fact, sparsity is not an appropriate metric because contourlets typically require 33% more coefficients than wavelets [42]. However, it is known that contourlets have better representations of lines and curves than wavelets.

2) Comparison Results: We synthetically create a gray-scaled image consisting of a triangle overlapping with an ellipse to simulate a disparity map. We choose the uniformly random sampling pattern S so that there is no bias caused by a particular sampling pattern.

As parameters are concerned, we set $\lambda_1 = 4 \times 10^{-5}$ and $\beta = 2 \times 10^{-3}$ for the single wavelet dictionary model ($L = 1$), and $\lambda_1 = 4 \times 10^{-5}$, $\lambda_2 = 2 \times 10^{-4}$ and $\beta = 2 \times 10^{-3}$ for the combined dictionary model ($L = 2$). The choices of these parameters are discussed in Section IV-C.

Using the proposed ADMM algorithm (See Section IV), we plot the performance of the reconstruction result as a function of the sampling ratio. For each point of the sampling ratio, we perform a Monte-Carlo simulation over 20 independent trials to reduce the fluctuation caused by the randomness in the sampling pattern. The result in Figure 2 indicates that the combined dictionary is consistently better than the wavelet dictionary alone. A snapshot of the result at $\xi = 0.1$ is shown
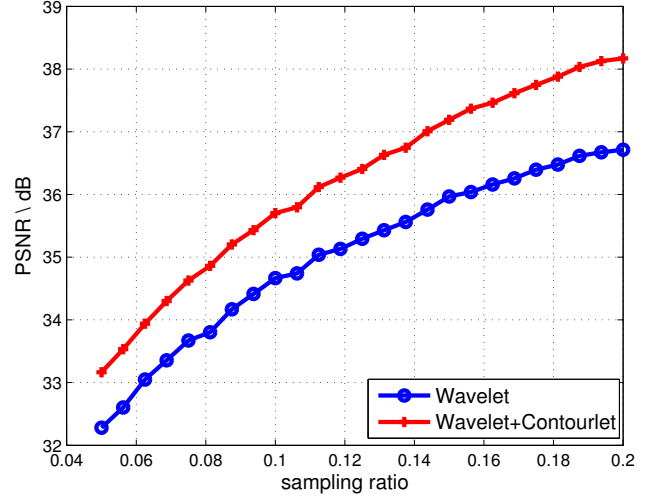

Fig. 2: ADMM reconstruction result as a function of sampling ratio ξ. Each point on the curves is averaged over 20 independent Monte-Carlo trials. The PSNR evaluates the performance of solving (11) using wavelet and wavelet+contourlet.

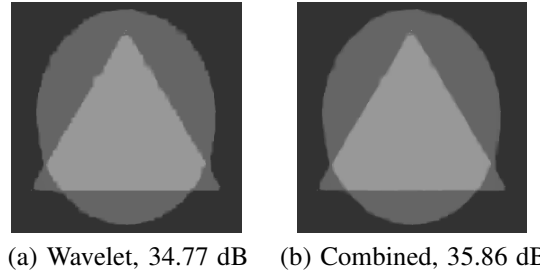(a) Wavelet, 34.77 dB

(b) Combined, 35.86 dB

Fig. 3: Snapshot of the comparison between wavelet dictionary and a combined wavelet-contourlet dictionary at $\xi = 0.1$.

in Figure 3. As observed, the reconstruction along the edges of the ellipse is better in the combined dictionary than using wavelet alone.

IV. RECONSTRUCTION ALGORITHM

In this section we present an alternating direction method of multipliers (ADMM) algorithm to solve (11). ADMM algorithms can be traced back to the proximal operators proposed by Moreau in the 60's [43], and later studied by Eckstein and Bertsekas [44] in the 90's. The application of ADMM to image deconvolution was first mentioned in [45]. For brevity we skip the introduction of the ADMM algorithm because comprehensive tutorials are easily accessible [46], [47]. Instead, we highlight the unique contributions of this paper, which include a particular operator splitting strategy and a multiscale implementation.

For notational simplicity we consider a single dictionary so that $L = 1$. Generalization to $L > 1$ is straight forward. Also, in our derivation we focus on the anisotropic total variation so that $\|x\|_{TV} = \|D_x x\|_1 + \|D_y x\|_1$. Extension to isotropic total variation follows the same idea as presented in [5].

A. ADMM and Operator Splitting

A central question about ADMM algorithms is which of the variables should be splitted so that the subsequent subproblems can be efficiently solved. Inspecting (11), we observe that there are many possible choices. For example, we could split the quadratic term in (11) by defining an auxiliary variable $\boldsymbol{u} = \boldsymbol{S}\boldsymbol{x}$, or we could keep the quadratic term without a split. In what follows, we present an overview of our proposed splitting method and discuss the steps in subsequent subsections.

We start the ADMM algorithm by introducing three auxiliary variables $\boldsymbol{r} = \boldsymbol{x}$, $\boldsymbol{u}_\ell = \boldsymbol{\Phi}_\ell \boldsymbol{x}$, and $\boldsymbol{v} = \boldsymbol{D}\boldsymbol{x}$. Consequently, we rewrite the optimization problem as

$$\begin{aligned} \underset{\boldsymbol{x}, \boldsymbol{r}, \boldsymbol{u}_\ell, \boldsymbol{v}}{\text{minimize}} \quad & \tfrac{1}{2}\|\boldsymbol{b} - \boldsymbol{S}\boldsymbol{r}\|^2 + \lambda_\ell \|\boldsymbol{W}_\ell \boldsymbol{u}_\ell\|_1 + \beta\|\boldsymbol{v}\|_1 \\ \text{subject to} \quad & \boldsymbol{r} = \boldsymbol{x}, \quad \boldsymbol{u}_\ell = \boldsymbol{\Phi}_\ell^T \boldsymbol{x}, \quad \boldsymbol{v} = \boldsymbol{D}\boldsymbol{x}. \end{aligned} \quad (12)$$

The ADMM algorithm is a computational procedure to find a stationary point of (12). The idea is to consider the augmented Lagrangian function defined as

$$\begin{aligned} \mathcal{L}&\left(\boldsymbol{x}, \boldsymbol{u}_\ell, \boldsymbol{r}, \boldsymbol{v}, \boldsymbol{w}, \boldsymbol{y}_\ell, \boldsymbol{z}\right) \\ &= \frac{1}{2}\|\boldsymbol{b} - \boldsymbol{S}\boldsymbol{r}\|^2 + \lambda_\ell \|\boldsymbol{W}_\ell \boldsymbol{u}_\ell\|_1 + \beta\|\boldsymbol{v}\|_1 \\ &\quad - \boldsymbol{w}^T(\boldsymbol{r} - \boldsymbol{x}) - \boldsymbol{y}_\ell^T\left(\boldsymbol{u}_\ell - \boldsymbol{\Phi}_\ell^T\boldsymbol{x}\right) - \boldsymbol{z}^T(\boldsymbol{v} - \boldsymbol{D}\boldsymbol{x}) \\ &\quad + \frac{\mu}{2}\|\boldsymbol{r} - \boldsymbol{x}\|^2 + \frac{\rho_\ell}{2}\|\boldsymbol{u}_\ell - \boldsymbol{\Phi}_\ell^T\boldsymbol{x}\|^2 + \frac{\gamma}{2}\|\boldsymbol{v} - \boldsymbol{D}\boldsymbol{x}\|^2. \end{aligned} \quad (13)$$

In (13), the vectors $\boldsymbol{w}$, $\boldsymbol{y}_\ell$ and $\boldsymbol{z}$ are the Lagrange multipliers; λ_ℓ and β are the regularization parameters, and μ, ρ_ℓ and γ are the internal half quadratic penalty parameters. The stationary point of the augmented Lagrangian function can be determined by solving the following sequence of subproblems

$$\begin{aligned} \boldsymbol{x}^{(k+1)} &= \underset{\boldsymbol{x}}{\operatorname{argmin}} \, \mathcal{L}\left(\boldsymbol{x}, \boldsymbol{u}_\ell^{(k)}, \boldsymbol{r}^{(k)}, \boldsymbol{v}^{(k)}, \boldsymbol{w}^{(k)}, \boldsymbol{y}_\ell^{(k)}, \boldsymbol{z}^{(k)}\right), \\ \boldsymbol{u}_\ell^{(k+1)} &= \underset{\boldsymbol{u}_\ell}{\operatorname{argmin}} \, \mathcal{L}\left(\boldsymbol{x}^{(k+1)}, \boldsymbol{u}_\ell, \boldsymbol{r}^{(k)}, \boldsymbol{v}^{(k)}, \boldsymbol{w}^{(k)}, \boldsymbol{y}_\ell^{(k)}, \boldsymbol{z}^{(k)}\right), \\ \boldsymbol{r}^{(k+1)} &= \underset{\boldsymbol{r}}{\operatorname{argmin}} \, \mathcal{L}\left(\boldsymbol{x}^{(k+1)}, \boldsymbol{u}_\ell^{(k+1)}, \boldsymbol{r}, \boldsymbol{v}^{(k)}, \boldsymbol{w}^{(k)}, \boldsymbol{y}_\ell^{(k)}, \boldsymbol{z}^{(k)}\right), \\ \boldsymbol{v}^{(k+1)} &= \underset{\boldsymbol{v}}{\operatorname{argmin}} \, \mathcal{L}\left(\boldsymbol{x}^{(k+1)}, \boldsymbol{u}_\ell^{(k+1)}, \boldsymbol{r}^{(k+1)}, \boldsymbol{v}, \boldsymbol{w}^{(k)}, \boldsymbol{y}_\ell^{(k)}, \boldsymbol{z}^{(k)}\right), \end{aligned}$$

and the Lagrange multipliers are updated as

$$\boldsymbol{y}_\ell^{(k+1)} = \boldsymbol{y}_\ell^{(k)} - \rho_\ell\left(\boldsymbol{u}_\ell^{(k+1)} - \boldsymbol{\Phi}_\ell^T\boldsymbol{x}^{(k+1)}\right), \quad (14a)$$

$$\boldsymbol{w}^{(k+1)} = \boldsymbol{w}^{(k)} - \mu\left(\boldsymbol{r}^{(k+1)} - \boldsymbol{x}^{(k+1)}\right), \quad (14b)$$

$$\boldsymbol{z}^{(k+1)} = \boldsymbol{z}^{(k)} - \gamma\left(\boldsymbol{v}^{(k+1)} - \boldsymbol{D}\boldsymbol{x}^{(k+1)}\right). \quad (14c)$$

We now discuss how each subproblem is solved.

B. Subproblems

1) x-subproblem: The $\boldsymbol{x}$-subproblem is obtained by dropping terms that do not involve $\boldsymbol{x}$ in (13). This yields

$$\begin{aligned} \boldsymbol{x}^{(k+1)} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \, & -\boldsymbol{w}^T(\boldsymbol{r} - \boldsymbol{x}) - \boldsymbol{y}_\ell^T\left(\boldsymbol{u}_\ell - \boldsymbol{\Phi}_\ell^T\boldsymbol{x}\right) \\ & - \boldsymbol{z}^T(\boldsymbol{v} - \boldsymbol{D}\boldsymbol{x}) + \frac{\mu}{2}\|\boldsymbol{r} - \boldsymbol{x}\|^2 \\ & + \frac{\rho_\ell}{2}\|\boldsymbol{u}_\ell - \boldsymbol{\Phi}_\ell^T\boldsymbol{x}\|^2 + \frac{\gamma}{2}\|\boldsymbol{v} - \boldsymbol{D}\boldsymbol{x}\|^2. \end{aligned} \quad (15)$$

Problem (15) can be solved by considering the first-order optimality condition, which yields a normal equation

$$\begin{aligned} \left(\rho_\ell \boldsymbol{\Phi}_\ell \boldsymbol{\Phi}_\ell^T + \mu\boldsymbol{I} + \gamma\boldsymbol{D}^T\boldsymbol{D}\right)&\boldsymbol{x}^{(k+1)} \\ = \boldsymbol{\Phi}_\ell\left(\rho_\ell\boldsymbol{u}_\ell - \boldsymbol{y}_\ell\right) &+ (\mu\boldsymbol{r} - \boldsymbol{w}) + \boldsymbol{D}^T\left(\gamma\boldsymbol{v} - \boldsymbol{z}\right). \end{aligned} \quad (16)$$

The matrix in (16) can be simplified as $(\rho_\ell + \mu)\boldsymbol{I} + \gamma\boldsymbol{D}^T\boldsymbol{D}$, because for any frame $\boldsymbol{\Phi}_\ell$, it holds that $\boldsymbol{\Phi}_\ell\boldsymbol{\Phi}_\ell^T = \boldsymbol{I}$. Now, since the matrix $\boldsymbol{D}^T\boldsymbol{D}$ is a circulant matrix, the matrix $(\rho_\ell + \mu)\boldsymbol{I} + \gamma\boldsymbol{D}^T\boldsymbol{D}$ is diagonalizable by the Fourier transform. This leads to a closed form solution as

$$\boldsymbol{x}^{(k+1)} = \mathcal{F}^{-1}\left[\frac{\mathcal{F}(\text{RHS})}{(\rho_\ell + \mu)\boldsymbol{I} + \gamma|\mathcal{F}(\boldsymbol{D})|^2}\right], \quad (17)$$

where RHS denotes the right hand side of (16), $\mathcal{F}(\cdot)$ denotes the 2D Fourier transform, $\mathcal{F}^{-1}(\cdot)$ denotes the 2D inverse Fourier transform, and $|\mathcal{F}(\boldsymbol{D})|^2$ denotes the magnitude square of the eigenvalues of the differential operator $\boldsymbol{D}$.

Remark 3: If we do not split the quadratic function $\|\boldsymbol{b} - \boldsymbol{S}\boldsymbol{x}\|^2$ using $\boldsymbol{r} = \boldsymbol{x}$, then the identity matrix $\mu\boldsymbol{I}$ in (16) would become $\mu\boldsymbol{S}^T\boldsymbol{S}$. Since $\boldsymbol{S}$ is a diagonal matrix containing 1's and 0's, the matrix $\rho_\ell\boldsymbol{\Phi}_\ell\boldsymbol{\Phi}_\ell^T + \mu\boldsymbol{S}^T\boldsymbol{S} + \gamma\boldsymbol{D}^T\boldsymbol{D}$ is not diagonalizable using the Fourier transform.

2) u_ℓ-subproblem: The $\boldsymbol{u}_\ell$-subproblem is given by

$$\min_{\boldsymbol{u}_\ell} \, \lambda_\ell\|\boldsymbol{W}_\ell\boldsymbol{u}_\ell\|_1 - \boldsymbol{y}_\ell^T\left(\boldsymbol{u}_\ell - \boldsymbol{\Phi}_\ell^T\boldsymbol{x}\right) + \frac{\rho_\ell}{2}\|\boldsymbol{u}_\ell - \boldsymbol{\Phi}_\ell^T\boldsymbol{x}\|^2. \quad (18)$$

Since $\boldsymbol{W}_\ell$ is a diagonal matrix, (18) is a separable optimization consisting of a sum of scalar problems. By using the standard shrinkage formula [5], one can show that the closed-form solution of (18) exists and is given by

$$\boldsymbol{u}_\ell^{(k+1)} = \max\left(\left|\boldsymbol{\alpha}_\ell + \frac{\boldsymbol{y}_\ell}{\rho_\ell}\right| - \frac{\lambda_\ell\widetilde{\boldsymbol{w}}_\ell}{\rho_\ell}, 0\right) \cdot \operatorname{sign}\left(\boldsymbol{\alpha}_\ell + \frac{\boldsymbol{y}_\ell}{\rho_\ell}\right), \quad (19)$$

where $\widetilde{\boldsymbol{w}}_\ell = \operatorname{diag}(\boldsymbol{W}_\ell)$ and $\boldsymbol{\alpha}_\ell = \boldsymbol{\Phi}_\ell^T\boldsymbol{x}$.

Remark 4: If we do not split using $\boldsymbol{u}_\ell = \boldsymbol{\Phi}_\ell^T\boldsymbol{x}$, then the $\boldsymbol{u}_\ell$-subproblem is not separable and hence the shrinkage formula cannot be applied. Moreover, if we split $\boldsymbol{u}_\ell = \boldsymbol{W}_\ell\boldsymbol{\Phi}_\ell^T\boldsymbol{x}$, *i.e.*, include $\boldsymbol{W}_\ell$, then the $\boldsymbol{x}$-subproblem will contain $\boldsymbol{\Phi}_\ell\boldsymbol{W}_\ell\boldsymbol{\Phi}_\ell^T$, which is not diagonalizable using the Fourier transform.

3) r-subproblem: The $\boldsymbol{r}$-subproblem is the standard quadratic minimization problem:

$$\min_{\boldsymbol{r}} \, \frac{1}{2}\|\boldsymbol{S}\boldsymbol{r} - \boldsymbol{b}\|^2 - \boldsymbol{w}^T(\boldsymbol{r} - \boldsymbol{x}) + \frac{\mu}{2}\|\boldsymbol{r} - \boldsymbol{x}\|^2. \quad (20)$$

Taking the first-order optimality yields a normal equation

$$\left(\boldsymbol{S}^T\boldsymbol{S} + \mu\boldsymbol{I}\right)\boldsymbol{r} = \left(\boldsymbol{S}^T\boldsymbol{b} + \boldsymbol{w} + \mu\boldsymbol{x}\right). \quad (21)$$

Since $\boldsymbol{S}$ is a diagonal binary matrix, (21) can be evaluated via an element-wise computation.

Remark 5: (21) shows that our splitting strategy of using $\boldsymbol{r} = \boldsymbol{x}$ is particularly efficient because $\boldsymbol{S}$ is a diagonal matrix. If $\boldsymbol{S}$ is a general matrix, *e.g.*, i.i.d. Gaussian matrix in [48], then solving (21) will be less efficient.

4) v-subproblem: The $\boldsymbol{v}$-subproblem is the standard total variation problem:

$$\min_{\boldsymbol{v}} \ \beta\|\boldsymbol{v}\|_1 - \boldsymbol{z}^T\left(\boldsymbol{v} - \boldsymbol{D}\boldsymbol{x}\right) + \frac{\gamma}{2}\|\boldsymbol{v} - \boldsymbol{D}\boldsymbol{x}\|^2. \qquad (22)$$

The solution is given by

$$\boldsymbol{v}^{(k+1)} = \max\left(\left|\boldsymbol{D}\boldsymbol{x} + \frac{\boldsymbol{z}}{\gamma}\right| - \frac{\beta}{\gamma}, 0\right) \cdot \mathrm{sign}\left(\boldsymbol{D}\boldsymbol{x} + \frac{\boldsymbol{z}}{\gamma}\right). \qquad (23)$$

The overall ADMM algorithm is shown in Algorithm 1.

Algorithm 1 ADMM Algorithm

Require: $\boldsymbol{b}, \boldsymbol{S}$
1: $\boldsymbol{x}^{(0)} = \boldsymbol{S}\boldsymbol{b}$, $\boldsymbol{u}_\ell^{(0)} = \boldsymbol{\Phi}_\ell^T \boldsymbol{x}^{(0)}$, $\boldsymbol{r}^{(0)} = \boldsymbol{x}^{(0)}$, $\boldsymbol{v}^{(0)} = \boldsymbol{D}\boldsymbol{x}^{(0)}$
2: **while** $\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\|_2 / \|\boldsymbol{x}^{(k)}\|_2 \geq \mathrm{tol}$ **do**
3:     Solve $\boldsymbol{x}$-subproblem by (17).
4:     Solve $\boldsymbol{u}_\ell$, $\boldsymbol{r}$ and $\boldsymbol{v}$ subproblems by (19), (21) and (23).
5:     Update multipliers by (14a), (14b) and (14c).
6: **end while**
7: **return** $\boldsymbol{x}^* \leftarrow \boldsymbol{x}^{(k+1)}$

C. Parameters

Parameter	Functionality	Values
λ_1	Wavelet sparsity	4×10^{-5}
λ_2	Contourlet sparsity	2×10^{-4}
β	Total variation	2×10^{-3}
ρ_1	Half quad. penalty for Wavelet	0.001
ρ_2	Half quad. penalty for Contourlet	0.001
μ	Half quad. penalty for $\boldsymbol{r} = \boldsymbol{x}$	0.01
γ	Half quad. penalty for $\boldsymbol{v} = \boldsymbol{D}\boldsymbol{x}$	0.1

TABLE I: Summary of Parameters.

The regularization parameters (λ_ℓ, β) and internal half quadratic penalty parameters (ρ_ℓ, μ, γ) are chosen empirically. Table I provides a summary of the parameters we use in this paper. These values are the typical values we found over a wide range of images and testing conditions. For detailed experiments of the parameter selection process, we refer the readers to our supplementary technical report in [49].

D. Convergence Comparison

Since (11) is convex, standard convergence proof of ADMM applies (c.f. [47]). Thus, instead of repeating the convergence theory, we compare our proposed algorithm with a subgradient algorithm proposed by Hawe et al. [24].

To set up the experiment, we consider the uniformly random sampling pattern $\boldsymbol{S}$ with sampling ratios $\xi = 0.1, 0.15, 0.2$. For both our algorithm and the subgradient algorithm proposed in [24], we consider a single wavelet dictionary using Daubechies wavelet "db2" with 2 decomposition levels. Other choices of wavelets are possible, but we observe that the difference is not significant.

Figure 4 shows the convergence results of our proposed algorithm and the subgradient algorithm. It is evident from the figure that the ADMM algorithm converges at a significantly
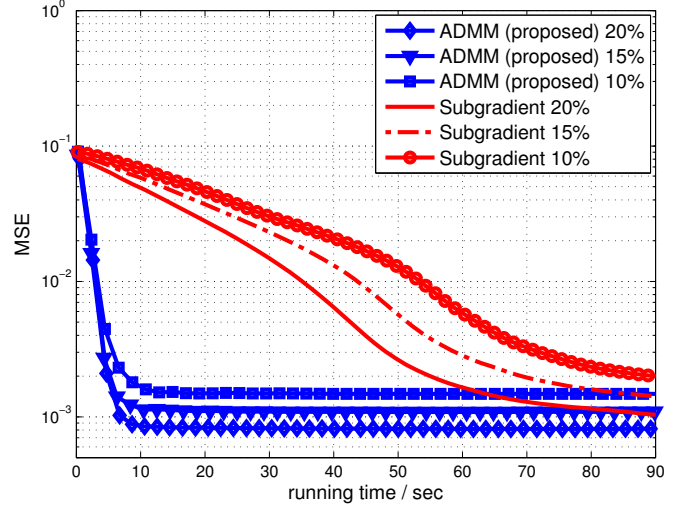

Fig. 4: Comparison of the rate of convergence between ADMM (proposed) and subgradient algorithms [24] for single wavelet dictionary. We used "Aloe" as a test image. The ADMM algorithm requires approximately 10 seconds to reach steady state. The subgradient algorithm requires more than $9\times$ running time than the ADMM algorithm to reach steady state.

faster rate than the subgradient algorithm. In particular, we see that the ADMM algorithm reaches a steady state in around 10 seconds, whereas the subgradient algorithm requires more than 90 seconds.

E. Multiscale ADMM

The ADMM algorithm shown in Algorithm 1 can be modified to incorporate a multiscale warm start. The idea works as follows.

First, given the observed data $\boldsymbol{b}$, we construct a multiscale pyramid $\{\boldsymbol{b}_q \mid q = 0, \ldots, Q-1\}$ of Q levels, with a scale factor of 2 across adjacent levels. Mathematically, by assuming without loss of generality that N is a power of 2, we define a downsampling matrix $\boldsymbol{A}_q$ at the qth level as

$$\boldsymbol{A}_q = [\boldsymbol{e}_1, \boldsymbol{0}, \boldsymbol{e}_2, \boldsymbol{0}, \ldots, \boldsymbol{0}, \boldsymbol{e}_{N/2^q}],$$

where $\boldsymbol{e}_k$ is the kth standard basis. Then, we define $\boldsymbol{b}_q$ as

$$\boldsymbol{b}_q = \boldsymbol{A}_q \boldsymbol{b}_{q-1}, \qquad (24)$$

for $q = 1, \ldots, Q-1$, and $\boldsymbol{b}_0 = \boldsymbol{b}$. Correspondingly, we define a pyramid of sampling matrices $\{\boldsymbol{S}_q \mid q = 0, \ldots, Q-1\}$, where

$$\boldsymbol{S}_q = \boldsymbol{A}_q \boldsymbol{S}_{q-1}, \qquad (25)$$

with the initial sampling matrix $\boldsymbol{S}_0 = \boldsymbol{S}$.

The above downsampling operation allows us to solve (11) at different resolution levels. That is, for each $q = 0, \ldots, Q-1$, we solve the problem

$$\boldsymbol{x}_q = \underset{\boldsymbol{x}}{\arg\min} \ \frac{1}{2}\|\boldsymbol{S}_q \boldsymbol{x} - \boldsymbol{b}_q\|_2^2 + \lambda_\ell \|\boldsymbol{W}_\ell \boldsymbol{\Phi}_\ell^T \boldsymbol{x}\|_1 + \beta\|\boldsymbol{x}\|_{TV}, \qquad (26)$$

where $\boldsymbol{\Phi}_\ell$ and $\boldsymbol{W}_\ell$ are understood to have appropriate dimensions.

Once $\boldsymbol{x}_q$ is computed, we feed an upsampled version of $\boldsymbol{x}_q$ as the initial point to the $(q-1)$th level's optimization. More specifically, we define an upsampling and averaging operation:

$$\boldsymbol{B}_q = \left[\boldsymbol{e}_1^T; \boldsymbol{e}_1^T; \boldsymbol{e}_2^T; \boldsymbol{e}_2^T; \ldots; \boldsymbol{e}_{N/2^q}^T; \boldsymbol{e}_{N/2^q}^T\right], \quad (27)$$

and we feed $\boldsymbol{x}_q$, the solution at the qth level, as the initial guess to the problem at the $(q-1)$th level:

$$\boldsymbol{x}_{q-1}^{(0)} = \boldsymbol{B}_q \boldsymbol{x}_q. \quad (28)$$

A pictorial illustration of the operations of $\boldsymbol{A}_q$ and $\boldsymbol{B}_q$ is shown in Figure 5. The algorithm is shown in Algorithm 2.
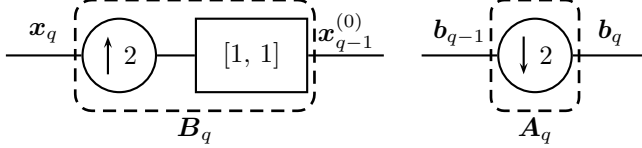
Fig. 5: Schematic diagram showing the operations of $\boldsymbol{A}_q$ and $\boldsymbol{B}_q$: $\boldsymbol{A}_q$ downsamples the observed data $\boldsymbol{b}_q$ by a factor of 2; $\boldsymbol{B}_q$ upsamples the solution $\boldsymbol{x}_q$ by a factor of 2, followed by a two-tap filter of impulse response $[1, 1]$.

Algorithm 2 Multiscale ADMM Algorithm
___
Require: $\boldsymbol{S}_0, \ldots, \boldsymbol{S}_{Q-1}$ and $\boldsymbol{b}_0, \ldots, \boldsymbol{b}_{Q-1}$
1: **for** $q = Q - 1$ **to** 0 **do**
2: $\quad \boldsymbol{x}_q = \text{ADMM}(\boldsymbol{b}_q, \boldsymbol{S}_q)$ with initial guess $\boldsymbol{x}_q^{(0)}$
3: $\quad$ Let $\boldsymbol{x}_{q-1}^{(0)} = \boldsymbol{B}_q \boldsymbol{x}_q$, if $q \geq 1$.
4: **end for**
5: Output $\boldsymbol{x} = \boldsymbol{x}_0$.

To validate the effectiveness of the proposed multiscale warm start, we compare the convergence rate against the original ADMM algorithm for a combined dictionary case. In Figure 6, we observe that the multiscale ADMM converges at a significantly faster rate than the original ADMM algorithm. More specifically, at a sampling ratio of 20%, the multiscale ADMM algorithm converges in 20 seconds whereas the original ADMM algorithm converges in 50 seconds which corresponds to a factor of 2.5 in runtime reduction. For fairness, both algorithms are tested under the same platform of MATLAB 2012b / 64-bit Windows 7 / Intel Core i7 / CPU 3.2GHz (single thread) / 12 GB RAM.

Remark 6: When propagating the qth solution, $\boldsymbol{x}_q$, to the $(q-1)$th level, we should also propagate the corresponding auxiliary variables $\boldsymbol{u}_\ell$, $\boldsymbol{r}$, $\boldsymbol{v}$ and the Lagrange multipliers $\boldsymbol{y}_\ell$, $\boldsymbol{w}$ and $\boldsymbol{z}$. The auxiliary variables can be updated according to $\boldsymbol{x}_{q-1}^{(0)}$ as $\boldsymbol{u}_{\ell,q-1}^{(0)} = \boldsymbol{\Phi}_\ell \boldsymbol{x}_{q-1}^{(0)}$, $\boldsymbol{r}_{q-1}^{(0)} = \boldsymbol{x}_{q-1}^{(0)}$, and $\boldsymbol{v}_{q-1}^{(0)} = \boldsymbol{D} \boldsymbol{x}_{q-1}^{(0)}$. For the Lagrange multipliers, we let $\boldsymbol{y}_{\ell,q-1}^{(0)} = \boldsymbol{B}_q \boldsymbol{y}_{\ell,q}$, $\boldsymbol{w}_{q-1}^{(0)} = \boldsymbol{B}_q \boldsymbol{w}_q$, and $\boldsymbol{z}_{q-1}^{(0)} = \boldsymbol{B}_q \boldsymbol{z}_q$.

Remark 7: The choice of the up/down sampling factor is not important. In our experiment, we choose a factor of 2 for simplicity in implementation. Other sampling factors such as $\sqrt{2}$ are equally applicable. Furthermore, the two-tap average filter $[1, 1]$ in Figure 5 can be replaced by any valid averaging filter. However, experimentally we find that other choices of filters do not make a significant difference comparing to $[1, 1]$.
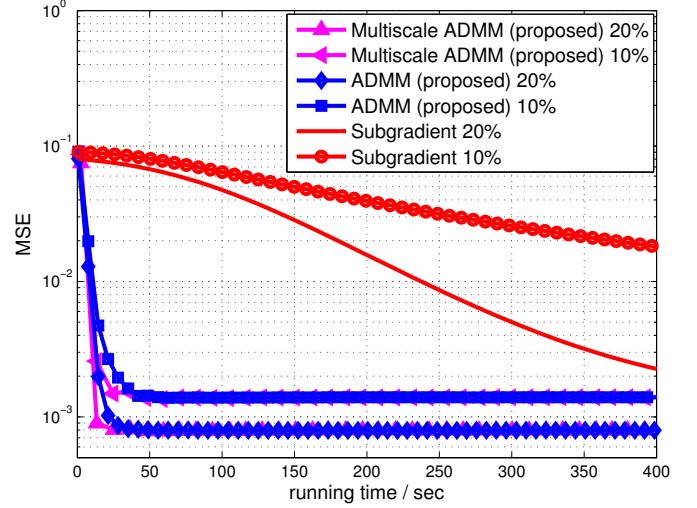

Fig. 6: Runtime comparison of original ADMM algorithm, multiscale ADMM algorithm and subgradient algorithm. All algorithms use the combined wavelet-contourlet dictionary. The testing image is "Aloe" and two sampling ratios (10% and 20%) are tested. $Q = 3$ multiscale levels are implemented in this experiment.

V. SAMPLING SCHEME

In the above sections, we assume that the sampling matrix $\boldsymbol{S}$ is given and is fixed. However, we have not yet discussed the design of the sampling probability $\{p_j\}_{j=1}^N$. The purpose of this section is to present an efficient design procedure.

A. Motivating Example

Before our discussion, perhaps we should first ask about what kind of sampling matrix $\boldsymbol{S}$ would work (or would not work). To answer this question, we consider an example shown in Figure 7. In Figure 7 we try to recover a simple disparity map consisting of an ellipse of constant intensity and a plain background. We consider three sampling patterns of approximately equal sampling ratios ξ: (a) a sampling pattern defined according to the magnitude of the disparity gradient; (b) an uniform grid with specified sampling ratio $\sqrt{\xi}$ along both directions; (c) a random sampling pattern drawn from an uniform distribution with probability ξ. The three sampling patterns correspondingly generate three sampled disparity maps. For each sampled disparity map, we run the proposed ADMM algorithm and record the reconstructed disparity map. In all experiments, we use a wavelet dictionary for demonstration.

The results in Figure 7 suggest a strong message: For a fixed sampling budget ξ, one should pick samples along gradients. However, the pitfall is that this approach is not practical for two reasons. First, the gradient of the disparity map is not available prior to reconstructing the disparity. Therefore, all gradient information can only be inferred from the color image. Second, the gradients of a color image could be very different from the gradients of the corresponding disparity map. Thus, inferring the disparity gradient from the color
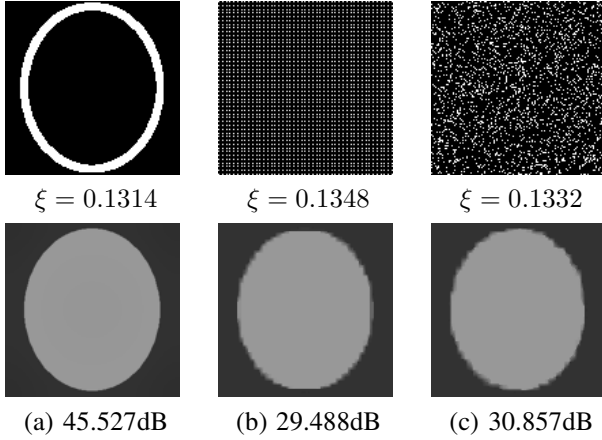
$\xi = 0.1314$ $\xi = 0.1348$ $\xi = 0.1332$

(a) 45.527dB   (b) 29.488dB   (c) 30.857dB

Fig. 7: Three sampling patterns and the corresponding reconstruction results using the proposed ADMM algorithm. Here, ξ denotes the actual sampling ratio. (a) Sampling along the gradient; (b) Sampling from a grid; (c) Sampling from an uniformly random pattern.

image gradient is a challenging task. In the followings, we present a randomized sampling scheme to address these two issues.

B. Oracle Random Sampling Scheme

We first consider an oracle situation where the gradients are assumed *known*. The goal is to see how much improvement one should expect to see.

Let $\boldsymbol{a} = [a_1, \ldots, a_N]^T$ be a vector denoting the magnitude of the ground truth disparity map's gradient. Given this oracle information about the disparity gradients, we consider a soft decision rule where a pixel is sampled with probability defined according to some function of $\{a_j\}_{j=1}^N$. Such a function is chosen based on the intuition that the sampled subset of gradients should carry the maximum amount of information compared to the full set of gradients. One way to capture this intuition is to require that the average gradient computed from *all* N samples is similar to the average gradient computed from a *subset* of ξN samples.

To be more precise, we define the average gradient computed from all N samples as

$$\mu \stackrel{\text{def}}{=} \frac{1}{N} \sum_{j=1}^N a_j. \tag{29}$$

Similarly, we define the average gradient computed from a random subset of ξN samples as

$$Y \stackrel{\text{def}}{=} \frac{1}{N} \sum_{j=1}^N \frac{a_j}{p_j} I_j, \tag{30}$$

where $\{I_j\}_{j=1}^N$ is a sequence of Bernoulli random variables with probability $\Pr[I_j = 1] = p_j$. Here, the division of a_j by p_j is to ensure that Y is unbiased, *i.e.*, $\mathbb{E}[Y] = \mu$.

From (29) and (30), minimizing the difference between Y and μ can be achieved by minimizing the variance $\mathbb{E}[(Y-\mu)^2]$.

(a) Greedy sampling   (b) Random sampling
35.5201 dB, $\xi = 0.1157$ 39.8976 dB, $\xi = 0.1167$

Fig. 8: Comparison between a deterministic sampling pattern by selecting samples greedily according to the magnitude of $\{a_j\}$, and a randomized sampling pattern using the proposed scheme.

Moreover, we observe that

$$\mathbb{E}\left[(Y-\mu)^2\right] = \frac{1}{N} \sum_{j=1}^N \frac{a_j^2}{p_j^2} \text{Var}\left[I_j\right] = \frac{1}{N} \sum_{j=1}^N a_j^2 \left(\frac{1-p_j}{p_j}\right),$$

where the last equality holds because $\text{Var}[I_j] = p_j(1 - p_j)$. Therefore, the optimal sampling probability $\{p_j\}_{j=1}^N$ can be found by solving the optimization problem

$$(P): \quad \underset{p_1,\ldots,p_N}{\text{minimize}} \quad \frac{1}{N} \sum_{j=1}^N \frac{a_j^2}{p_j}$$

$$\text{subject to} \quad \frac{1}{N} \sum_{j=1}^N p_j = \xi, \text{ and } 0 \le p_j \le 1,$$

of which the solution is given by [50, Lemma 2]

$$p_j = \min(\tau a_j, 1), \tag{31}$$

where τ is the root of the equation

$$g(\tau) \stackrel{\text{def}}{=} \sum_{j=1}^N \min(\tau a_j, 1) - \xi N. \tag{32}$$

It is interesting to compare this new random sampling scheme versus a greedy sampling scheme by picking the ξN pixels with the largest gradients. Figure 8 shows the result. For the greedy sampling scheme, we first compute the gradient of the disparity map $\nabla \boldsymbol{x} \stackrel{\text{def}}{=} \sqrt{(\boldsymbol{D}_x \boldsymbol{x})^2 + (\boldsymbol{D}_y \boldsymbol{x})^2}$ and threshold it to obtain a set of samples $\Omega \stackrel{\text{def}}{=} \{j \mid [\nabla \boldsymbol{x}]_j > \alpha \|\nabla \boldsymbol{x}\|_\infty\}$, where $\alpha = 0.1$ is the threshold. The actual sampling ratio is then $|\Omega|/N$. For the randomized scheme, we let $\boldsymbol{a} = \nabla \boldsymbol{x}$ and we compute p_j according to (31). In this particular example, we observe that the randomized sampling scheme achieves a PSNR improvement of more than 4 dB.

C. Practical Random Sampling Scheme

We now present a practically implementable sampling scheme. The challenge that we have to overcome is that the gradient information of the disparity is not available. Therefore, we propose the following two-stage sampling process.

Our proposed sampling scheme consists of two stages - a pilot stage to obtain a rough estimate of the disparity, and a refinement stage to improve the disparity estimate. In the first step pilot stage, we pick $\xi N/2$ samples (*i.e.*, half of the desired number of samples) using an uniformly random

sampling pattern. This gives a sampling pattern $\{I_j^{(1)}\}_{j=1}^N$, where the superscript denotes the first stage. Correspondingly, we have a sampling matrix $\boldsymbol{S}^{(1)}$ and the sampled data $\boldsymbol{b}^{(1)}$. Given $\boldsymbol{S}^{(1)}$ and $\boldsymbol{b}^{(1)}$, we apply the ADMM algorithm to obtain a pilot estimate $\boldsymbol{x}^{(1)}$.

In the second stage, we use the pilot estimate $\boldsymbol{x}^{(1)}$ as a guide to compute the gradient $\nabla \boldsymbol{x}^{(1)}$. By (31), this suggests that the optimal sampling probability is $p_j = \min(\tau[\nabla \boldsymbol{x}^{(1)}]_j, 1)$. However, in order to ensure that the $\xi N/2$ samples picked at the second stage *do not overlap* with those picked in the first stage, instead of letting $p_j = \min(\tau[\nabla \boldsymbol{x}^{(1)}]_j, 1)$, we let $p_j = \min(\tau a_j, 1)$, where

$$a_j = \begin{cases} [\nabla \boldsymbol{x}^{(1)}]_j, & \text{if } I_j^{(1)} = 0, \\ 0, & \text{if } I_j^{(1)} = 1. \end{cases} \tag{33}$$

In words, a_j defined by (33) forces $p_j = 0$ when the jth pixel is picked in the first step. Thus, the non-zero entries of $\{I_j^{(1)}\}$ and $\{I_j^{(2)}\}$ are mutually exclusive, and hence we can now apply the ADMM algorithm to recover $\boldsymbol{x}^{(2)}$ from $\boldsymbol{S}_1 + \boldsymbol{S}_2$ and $\boldsymbol{b}_1 + \boldsymbol{b}_2$. The overall method is summarized in Algorithm 3.

Algorithm 3 Two-Stage Algorithm

1: Input: $N, \xi, \boldsymbol{b}$
2: Output: $\boldsymbol{x}^{(2)}$

3: **Stage 1:**
4:    Let $I_j^{(1)} = 1$ with probability $\xi/2$, for $j = 1, \ldots, N$.
5:    Define $\boldsymbol{S}^{(1)}$ and $\boldsymbol{b}^{(1)}$ according to $\{I_j^{(1)}\}$.
6:    Compute $\boldsymbol{x}^{(1)} = \text{ADMM}(\boldsymbol{S}^{(1)}, \boldsymbol{b}^{(1)})$.

7: **Stage 2:**
8:    Compute $\nabla \boldsymbol{x}^{(1)}$.
9:    For $j = 1, \ldots, N$, define $a_j = \begin{cases} [\nabla \boldsymbol{x}^{(1)}]_j, & \text{if } I_j^{(1)} = 0, \\ 0, & \text{if } I_j^{(1)} = 1. \end{cases}$
10:   Compute τ such that $\sum_{j=1}^N \min\{\tau a_j, 1\} = N\xi/2$.
11:   Let $p_j = \min\{\tau a_j, 1\}$, for $j = 1, \ldots, N$.
12:   Let $I_j^{(2)} = 1$ with probability p_j, for $j = 1, \ldots, N$.
13:   Define $\boldsymbol{S}^{(2)}$ and $\boldsymbol{b}^{(2)}$ according to $\{I_j^{(2)}\}$.
14:   Compute $\boldsymbol{x}^{(2)} = \text{ADMM}(\boldsymbol{S}^{(1)} + \boldsymbol{S}^{(2)}, \boldsymbol{b}^{(1)} + \boldsymbol{b}^{(2)})$.

D. Further Improvement by PCA

The two-stage sampling procedure can be further improved by utilizing the prior information of the color image. The intuition is that since both color image and disparity map are captured from the same scene, strong gradients in the disparity map should align with those in the color image. However, since a color image typically contains complex gradients which are irrelevant to the disparity reconstruction, it is important to filter out these unwanted gradients while preserving the important ones. To this end, we consider the following patch-based principal component analysis.

Given a color image $\boldsymbol{y} \in \mathbb{R}^N$, we construct a collection of patches $\{\boldsymbol{y}_j\}_{j=1}^N$ where $\boldsymbol{y}_j \in \mathbb{R}^d$ denotes a vectorization
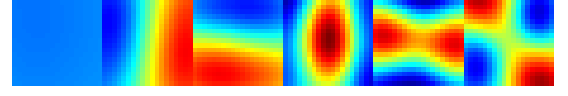

Fig. 9: The first 6 eigenvectors of the data matrix $\boldsymbol{Y}\boldsymbol{Y}^T$, where $\boldsymbol{Y}$ is obtained from the color image corresponding to Figure 8. In this example we set the patch size as 19×19 so that $d = 361$. The range of the color index of this figure is $[-0.1, 0.1]$.

of the jth patch of size $\sqrt{d} \times \sqrt{d}$ centered at pixel j of the image. For patches centered at the corners or boundaries of the image, we apply a symmetrical padding to make sure that their sizes are $\sqrt{d} \times \sqrt{d}$. This will give us a total of N patches.

Next, we form a data matrix $\boldsymbol{Y} \stackrel{\text{def}}{=} [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N]$. This data matrix leads to a principal component decomposition as

$$\boldsymbol{Y}\boldsymbol{Y}^T = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T, \tag{34}$$

where $\boldsymbol{U}$ is the eigenvector matrix, and $\boldsymbol{\Lambda}$ is the eigenvalue matrix. Given $\boldsymbol{U}$, we can compute $\boldsymbol{u}_i^T \boldsymbol{y}_j$, *i.e.*, the projection of a patch $\boldsymbol{y}_j$ onto the subspace spanned by an eigenvector $\boldsymbol{u}_i$. If we view $\boldsymbol{u}_i$ as a finite impulse response filter, then the projection is equivalent to a filtering.

The structure of the filters deserves a closer look. In Figure 9 we show the 6 leading eigenvectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_6$. It can be seen that except for the first eigenvector $\boldsymbol{u}_1$ (which is a constant vector), all remaining eigenvectors $\boldsymbol{u}_2, \ldots, \boldsymbol{u}_d$ are in the form of differential operators, with different orders and orientations. Moreover, these filters are *bandpass* filters, which suggests that various gradients of the color image can be extracted by the filtering. Consequently, if one would like to extract major gradients while rejecting gradients of the textures, the following filtered signal can be considered:

$$a_j = \sum_{i=2}^{d'} |\boldsymbol{u}_i^T \boldsymbol{y}_j|, \quad j = 1, \ldots, N, \tag{35}$$

where $d' < d$ is a tunable parameter (which was set to $d' = 16$ for $d = 49$ in this paper). Here, the absolute value in (35) is used to get the magnitude of $\langle \boldsymbol{u}_i, \boldsymbol{y}_j \rangle$, as a_j must be a non-negative number.

To see how this PCA concept can be incorporated into our two-stage sampling scheme, we make the following observations. First, the uniform sampling in Stage-1 can well be replaced by the PCA approach. In particular, instead of setting $I_j^{(1)} = 1$ with probability $\xi/2$, we can define a_j according to (35), and let $p_j = \min(\tau a_j, 1)$ for τ being the root of (32). Consequently, we let $I_j^{(1)} = 1$ with probability p_j.

In Stage-2, since we have already had a pilot estimate of the disparity map, it is now possible to replace $\boldsymbol{Y}$ in (34) by a data matrix $\boldsymbol{X} = [\boldsymbol{x}_1^{(1)}, \ldots, \boldsymbol{x}_N^{(1)}]$, where each $\boldsymbol{x}_j^{(1)}$ is a d-dimensional patch centered at the jth pixel of $\boldsymbol{x}^{(1)}$. Thus, instead of setting $a_j = [\nabla \boldsymbol{x}^{(1)}]_j$ in (33), we can set $a_j = \sum_{i=2}^{d'} |\langle \boldsymbol{u}_i, \boldsymbol{x}_j^{(1)} \rangle|$ using (35). The advantage of this new a_j is that it softens the sampling probability at the object boundaries to a neighborhood surrounding the boundary. This reduces the risk of selecting irrelevant samples because of a bad pilot estimate.

(a) Uniform random	(b) Uniform grid	(c) Proposed w/o PCA	(d) Proposed w/ PCA
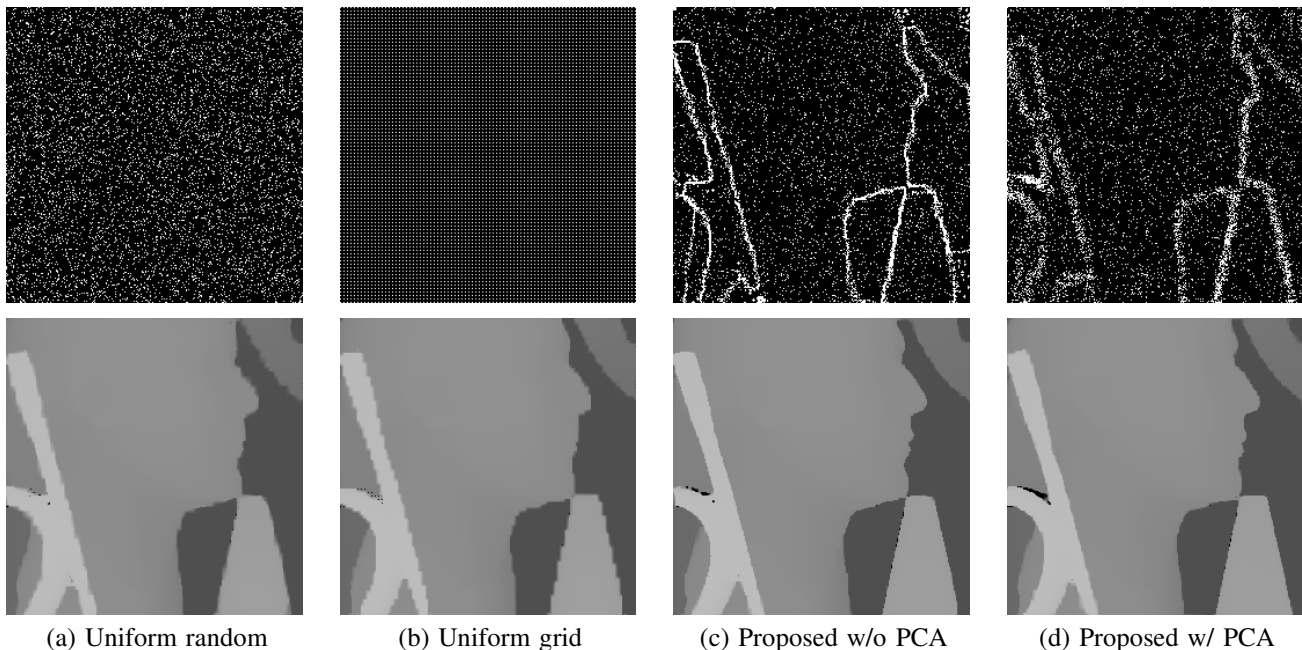
Method	Actual Sampling Ratio	Average PSNR / dB	Standard Deviation
Uniform random	0.1001	29.7495	0.3768
Uniform grid	0.1128	30.2726	0.0000
Proposed w/o PCA	0.1000	32.4532	0.8962
Proposed w/ PCA	0.1002	**33.7707**	1.0435

Fig. 10: Comparison between four sampling patterns. (a) Uniformly random sampling pattern; (b) Uniform grid; (c) Proposed two-stage sampling without PCA improvement; (d) Proposed two-stage sampling with PCA improvement. For the two-stage sampling in (c)-(d), we pick $\xi N/2$ uniformly random samples in stage 1, and pick the remaining $\xi N/2$ samples according to the pilot estimate from Stage 1. We conduct a Monte-Carlo simulation with 32 independent trials. The averages of PSNRs are presented in the Table.

E. Comparisons

As a comparison between sampling patterns, we consider a disparity map shown in Figure 10. Setting $\xi = 0.1$ (*i.e.*, 10%), we study four sampling patterns including two versions of our proposed two-stage method. We conduct a Monte-Carlo simulation by repeating 32 independent trials, and average the PSNRs. The results shown in Figure 10(c) are generated using the original two-stage sampling scheme without PCA improvement, whereas the results shown in Figure 10(d) are generated using an improved two-stage sampling scheme where the first stage is uniform and the second stage is PCA. These results indicate that for the same sampling ratio ξ, the choice of the sampling pattern has some strong influence to the reconstruction quality. For example, as compared to both uniform random sampling and grid sampling, the original two-stage sampling has about 2.44 dB improvement, and can be further improved by almost 3.76 dB using the PCA idea.

VI. EXPERIMENTAL RESULTS

In this section we present additional results to illustrate the performance of the proposed method.

A. Synthetic Data

We first compare the proposed algorithm with existing methods on the Middlebury dataset[1] where ground truth disparities are available. We consider two versions of the proposed algorithm: "Proposed WT+CT Grid" and "Proposed WT+CT 2-Stage". "Proposed WT+CT Grid" is the ADMM algorithm presented in Section IV using both wavelet and contourlet bases. Here, "Grid" refers to using a deterministic uniform grid sampling pattern and "2-stage" refers to using the 2-stage randomized sampling scheme presented in Section V. We use Daubechies wavelet "db2" with 2 decomposition levels for wavelet dictionary, and we set "bior9-7" wavelet function with [5 6] directional decompositions for contourlet dictionary.

We also compare our method with [24], which has three differences from ours: (1) [24] uses a subgradient descent algorithm whereas we use an ADMM algorithm; (2) [24] considers only a wavelet basis whereas we consider a combined wavelet-contourlet basis; (3) [24] uses a combination of canny edges and uniformly random samples whereas we use a principled design process to determine samples.

In this experiment we do not compare with depth super resolution algorithms, *e.g.*, [18], [51], [52]. These methods re-

---

[1] http://vision.middlebury.edu/stereo/data/

quire a color image to guide the actual reconstruction process, which is different from what is presented here because we only use the color image for designing the sampling pattern. As a reference of these methods, we show the results of a bicubic interpolation using uniform grid sampling pattern.

Table II shows the PSNR values of various methods at different sampling ratios and sampling methods. It is clear that "Proposed WT+CT 2-Stage" outperforms the other methods by a significant margin. Moreover, as the sampling ratio increases, the PSNR gain of "Proposed WT+CT 2-Stage" is more prominent than that of other methods. For example, increasing from 5% to 25% for "Art", "Proposed WT+CT 2-Stage" demonstrates an 18 dB PSNR improvement whereas bicubic only demonstrates 3 dB improvement.

It is also instructive to compare the percentage of bad pixels (% Bad Pixel), which is a popular metric to measure the quality of disparity estimates [53]. Given a threshold $\tau > 0$, the percentage of bad pixels is defined as

$$\% \text{ Bad Pixel} \overset{\text{def}}{=} \frac{1}{N} \sum_{j=1}^{N} \left(|\widehat{x}_j - x_j^*| > \tau \right), \qquad (36)$$

where $\widehat{x}$ is the reconstructed disparity and x^* is the ground truth disparity. Percentage of bad pixels can be considered as an absolute difference metric as compared to the mean squared metric of PSNR.

Table III shows the percentage of bad pixels of various methods at different sampling ratios and sampling methods. The results indicate that "Proposed WT+CT 2-Stage" has a relatively higher % Bad Pixel at $\tau = 1$ than other methods, but has a lower % Bad Pixel at $\tau = 2$ and $\tau = 3$. This result suggests that most of the errors of "Proposed WT+CT 2-Stage" are *small* and there are very few outliers. In contrast, bicubic grid (for example) has a low % Bad Pixel at $\tau = 1$ but high % Bad Pixel at $\tau = 2$ and $\tau = 3$. This implies that a significant portion of the bicubic results has large error. Intuitively, the results suggest that in the bicubic case, some strong edges and corners are completely missed, whereas these information are kept in "Proposed WT+CT 2-Stage".

Finally, we show the performance of the proposed algorithm towards additive i.i.d. Gaussian noise. The purpose of this experiment is to demonstrate the sensitivity and robustness of the algorithm in the presence of noise. While in reality the noise in disparity estimates is not i.i.d. Gaussian, the result presented here serves as a reference for the algorithm's performance. A more realistic experiment on real data will be illustrated in the next subsection.

The results are shown in Figure 11. Using "Bicubic Grid" as the baseline, we observe that "Proposed WT+CT 2-Stage" on average has 5.79 dB improvement, "Proposed WT+CT Grid" has 3.60 dB improvement, whereas "[24] Grid" has only 3.02 dB improvement. This provides a good indicator of the robustness of the proposed methods.

B. Real Data

In this experiment we study the performance of the proposed algorithm for real data. The top left part of Figure 12 shows a snapshot of a stereo video (with resolution 320×240, 30
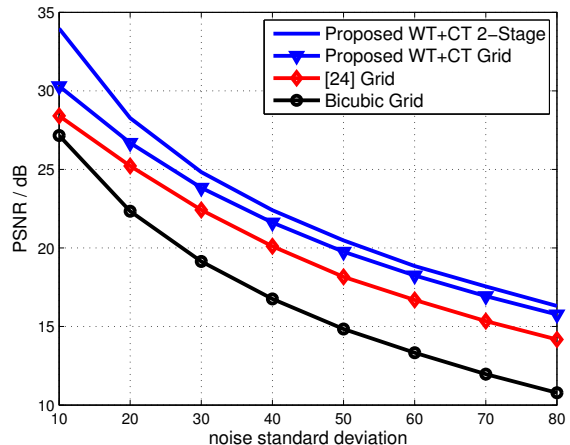

Fig. 11: Comparison of reconstruction performance with noisy samples. We use "Art" disparity map as a test image, and set $\xi = 0.2$.

fps). For this video sequence, we apply the block matching algorithm by Lee et al. [54] to obtain the initial disparity estimates. However, instead of computing the *full disparity map*, we only compute 10% of the disparity pixel values and use the proposed reconstruction algorithm to recover the dense disparity map. The 10% samples are selected according to the two stages of "Proposed WT+CT 2-Stage". In the first stage, we select the locations of the 5% samples using our oracle random sampling scheme with PCA improvement applied to the color image. A pilot estimate of the disparity is thus computed and the remaining 5% samples can be located according to the second stage of "Proposed WT+CT 2-Stage". The results shown in the middle row of Figure 12 illustrate that the "Proposed WT+CT 2-Stage" generates the closest disparity maps compared to an ideal dense estimate.

In addition to real video sequences, we also test the proposed algorithm on a stereo system we developed. The system consists of a low cost stereo camera with customized block matching algorithms. The bottom row of Figure 12 shows the results of the reconstructed disparity maps. Referring to the results of "[24] Grid" and "Bicubic Grid", we note that there are serious stair-like artifacts located at object boundaries. In contrast, the two proposed methods in general produce much smoother object boundaries, thanks to the superior modeling and the optimized sampling scheme. More interestingly, we observe that "Proposed WT+CT 2-Stage" indeed removes some undesirable noisy estimates in the recovered disparity maps. This shows that the proposed method could potentially further developed as a depth enhancement method.

VII. CONCLUSION

We proposed a framework to estimate dense depth maps from sparse samples. The new framework consists of three components: Representation, algorithm, and sampling. First, a combined wavelet-contourlet dictionary is proposed to encode the depth maps with a sparse set of bases. Consequently, we proposed an alternating direction method of multipliers (ADMM) with a multi-scale warm start to estimate the depth maps. The ADMM algorithm achieves faster rate of conver-

TABLE II: Comparisons of reconstruction algorithms in terms of PSNR. We put N/A when the algorithm does not converge in 1000 iterations.

Disparity Name	Method Algorithm / Sampling Strategy	Percentage of Samples / PSNR (dB)				
		5%	10%	15%	20%	25%
Aloe	Proposed WT+CT 2-Stage	27.5998	**31.3877**	**33.3693**	**36.4102**	**38.6265**
	Proposed WT+CT Grid	25.3236	28.9052	30.0940	31.2956	32.3548
	[24] Grid	25.1248	27.8941	28.9504	30.2371	31.6646
	Bicubic Grid	**27.8899**	29.3532	30.1019	31.0031	31.8908
Art	Proposed WT+CT 2-Stage	**30.8669**	**34.1495**	**37.2801**	**42.9706**	**48.0002**
	Proposed WT+CT Grid	27.5176	28.9528	30.8371	32.5150	33.7126
	[24] Grid	27.0300	N/A	N/A	N/A	N/A
	Bicubic Grid	29.1550	30.3536	31.1098	31.9473	32.8366
Baby	Proposed WT+CT 2-Stage	**39.6978**	**44.8958**	**48.6631**	**52.5000**	**52.0031**
	Proposed WT+CT Grid	34.4421	36.7965	37.6708	39.0504	40.0689
	[24] Grid	33.6627	35.3166	36.2522	37.4513	38.7670
	Bicubic Grid	34.8368	36.2385	37.1749	37.5973	38.3961
Dolls	Proposed WT+CT 2-Stage	**29.5087**	**32.5336**	**33.9974**	**36.2741**	**37.6527**
	Proposed WT+CT Grid	28.4858	29.0453	30.0949	30.8123	31.6725
	[24] Grid	28.4959	N/A	N/A	N/A	32.0521
	Bicubic Grid	29.0612	30.0475	30.4374	31.0053	31.8800
Moebius	Proposed WT+CT 2-Stage	**31.0663**	**35.1060**	**37.7626**	**39.9225**	**41.8933**
	Proposed WT+CT Grid	27.6882	28.7245	29.8527	31.1663	32.2399
	[24] Grid	27.6851	28.7973	N/A	N/A	32.0990
	Bicubic Grid	28.3987	29.9338	30.6607	30.9427	32.0143
Rocks	Proposed WT+CT 2-Stage	**30.7662**	**35.3975**	**37.5056**	**40.4494**	**42.5089**
	Proposed WT+CT Grid	25.5924	29.0848	30.4766	31.2311	32.9218
	[24] Grid	25.4444	28.7973	29.5364	30.2058	32.1672
	Bicubic Grid	28.7241	30.4212	30.7552	31.6722	32.6706

TABLE III: Comparisons of reconstruction algorithms in terms of % Bad Pixel.

Disparity Name	Method Algorithm Sampling Strategy	% of Bad Pixels [$\tau = 1$]					% of Bad Pixels [$\tau = 2$]					% of Bad Pixels [$\tau = 3$]				
		Percentage of Samples					Percentage of Samples					Percentage of Samples				
		5%	10%	15%	20%	25%	5%	10%	15%	20%	25%	5%	10%	15%	20%	25%
Aloe	Prop. WT+CT 2-Stage	41.47	21.37	14.00	8.85	5.81	**20.03**	**7.15**	**3.70**	**1.99**	**1.11**	**13.42**	**4.80**	**2.52**	**1.43**	**0.79**
	Prop. WT+CT Grid	36.88	22.96	15.61	11.62	8.69	21.16	10.11	6.87	5.17	3.92	15.80	7.62	5.55	4.25	3.27
	[24] Grid	31.44	**17.65**	**11.58**	**8.39**	**5.79**	20.12	8.87	6.08	4.73	3.56	14.73	6.97	5.03	4.01	3.09
	Bicubic Grid	**31.23**	23.37	18.62	15.88	13.39	23.51	17.49	13.78	11.96	10.04	19.40	14.47	11.51	9.96	8.30
Baby	Prop. WT+CT 2-Stage	28.00	12.37	5.72	**2.67**	**1.31**	9.95	**2.31**	**0.39**	**0.13**	**0.07**	**3.69**	**0.64**	**0.16**	**0.03**	**0.01**
	Prop. WT+CT Grid	15.80	8.27	5.80	4.12	3.14	**6.31**	3.01	2.22	1.58	1.22	4.25	2.30	1.77	1.31	1.05
	[24] Grid	12.31	**6.02**	**3.93**	2.71	1.86	6.44	2.73	1.94	1.47	1.10	4.21	2.09	1.55	1.21	0.93
	Bicubic Grid	**12.22**	8.53	6.54	5.59	4.58	7.89	5.63	4.34	3.73	3.10	6.24	4.42	3.51	3.00	2.41
Rocks	Prop. WT+CT 2-Stage	25.90	10.67	6.27	**3.55**	**2.19**	8.26	**2.26**	**0.93**	**0.41**	**0.22**	**4.75**	**1.22**	**0.51**	**0.21**	**0.10**
	Prop. WT+CT Grid	20.67	11.74	8.03	5.79	4.44	**7.64**	4.12	2.93	2.34	1.72	5.16	3.01	2.27	1.88	1.43
	[24] Grid	16.36	**9.09**	**6.10**	4.38	3.00	8.33	4.02	2.86	2.24	1.62	5.52	2.93	2.19	1.76	1.26
	Bicubic Grid	**15.32**	11.51	9.36	7.88	6.59	10.20	7.95	6.46	5.26	4.61	8.28	6.51	5.24	4.42	3.76

gence than the existing subgradient descent algorithms. Based on the ADMM algorithm, we also proposed a randomized sampling scheme to optimally choose the samples for a given sampling budget. The new framework can be applied to many depth data processing tasks including acquisition, compression, and enhancement. Future work shall be focused on extending the methods to space-time data volume to further improve consistency of the depth map estimates.

REFERENCES

[1] M. A. Lefsky, W. B. Cohen, G. G. Parker, and D. J. Harding, "LIDAR remote sensing of above-ground biomass in three biomes," *Global Ecology and Biogeography*, vol. 11, pp. 393–399, Oct. 2002.

[2] S. Agarwal, N. Snavely, I. Simon, S.M. Seitz, and R. Szeliski, "Building Rome in a day," in *Proc. IEEE Int. Conf. Computer Vision (ICCV'09)*, Sep. 2009, pp. 72–79.

[3] S. Burion, "Human detection for robotic urban search and rescue," M.S. thesis, Carnegie Mellon Univ., 2004, available at http://www.cs.cmu.edu/afs/cs/project/retsina-31/www/Report/Final%20Report.pdf.

[4] R. Khoshabeh, J. Juang, M.A. Talamini, and T.Q. Nguyen, "Multiview glasses-free 3-D laparoscopy," *IEEE Trans. Bio. Eng.*, vol. 59, no. 10, pp. 2859–2865, Oct. 2012.

[5] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, "An augmented Lagrangian method for total variation video restoration," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3097–3111, Nov. 2011.

[6] S. Foix, G. Alenya, and C. Torras, "Lock-in time-of-flight (ToF) cameras: A survey," *IEEE Sensors Journal*, vol. 11, no. 9, pp. 1917–1926, Sep. 2011.

[7] B. Schwarz, "LIDAR: Mapping the world in 3D," *Nature Photonics*, vol. 4, pp. 429–430, Jul. 2010.

[8] C. Niclass, M. Soga, H. Matsubara, S. Kato, and M. Kagami, "A 100-m range 10-frame/s 340 × 96-pixel time-of-flight depth sensor in 0.18-μm cmos," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 2, pp. 559–572, Feb. 2013.

[9] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang, "On building an accurate stereo matching system on graphics hardware," in *Proc.*
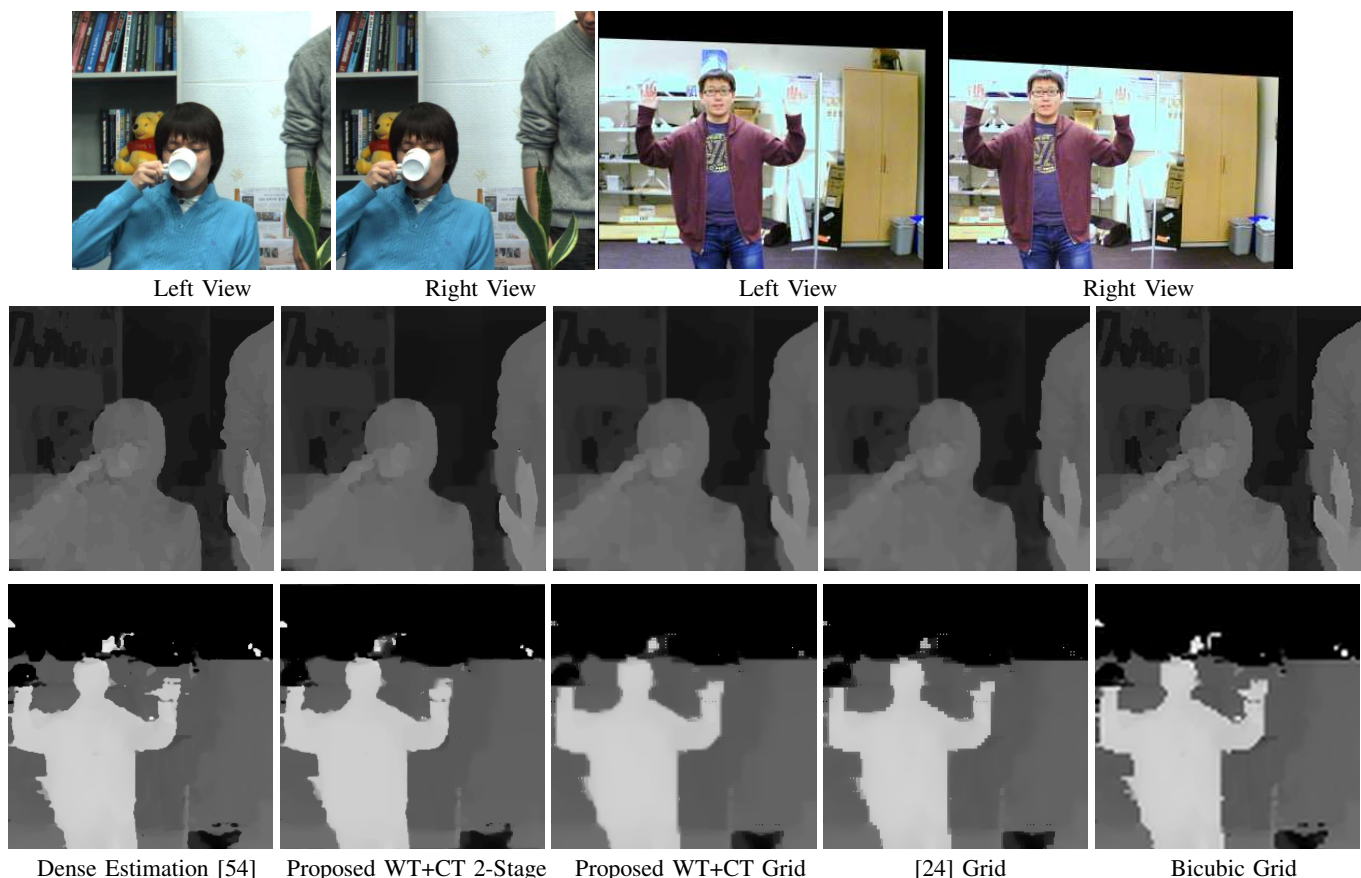
Fig. 12: Examples of disparity map reconstruction from 10% measured samples using real data. [Top] Left and right view images of the "Newspaper" dataset, and a sequence captured by a stereo system we developed. [Middle] The reconstructed disparity maps of "Newspaper". [Bottom] The reconstructed disparity maps of our sequence. For the reconstructed disparity maps, we show the zoom-in results of size 256×256 for better visualization. Methods under comparisons include: a dense disparity estimation [54] to acquire initial estimate; "Proposed WT+CT 2-Stage" which applies the 2-Stage randomized scheme to determine sampling locations; "Proposed WT+CT Grid" which picks samples from a uniform grid; "[24] Grid" which applies a subgradient algorithm to samples picked from a uniform grid; "Bicubic Grid" which applies bicubic interpolation to samples picked from a uniform grid.

IEEE Int. Conf. Computer Vision (ICCV'11), Nov. 2011, pp. 467–474.

[10] A. Klaus, M. Sormann, and K. Karner, "Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure," in *Proc. IEEE Int. Conf. on Pattern Recognition (ICPR'06)*, Aug. 2006, vol. 3, pp. 15–18.

[11] Z. Wang and Z. Zheng, "A region based stereo matching algorithm using cooperative optimization," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'08)*, Jun. 2008, pp. 1–8.

[12] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nister, "Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 31, no. 3, pp. 492–504, Mar. 2009.

[13] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *in Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'97)*, Jun. 1997, pp. 1106–1112.

[14] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *Proc. IEEE Int. Conf. Computer Vision (ICCV'99)*, Sep. 1999, vol. 1, pp. 666–673.

[15] R. S. Feris, J. Gemmell, K. Toyama, and V. Kruger, "Hierarchical wavelet networks for facial feature localization," in *Proc. IEEE Int. Conf. Automatic Face and Gesture Recognition (FG'02)*, May 2002, pp. 118–123.

[16] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'04)*, Jun. 2004, vol. 2, pp. 506–513.

[17] J. Diebel and S. Thrun, "An application of Markov random field to range sensing," in *Advances in Neural Info. Process. System (NIPS'05)*, Dec. 2005, pp. 291–298.

[18] Q. Yang, R. Yang, J. Davis, and D. Nister, "Spatial-depth super resolution for range images," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'07)*, Jun. 2007, pp. 1–8.

[19] J. Li, T. Xue, L. Sun, and J. Liu, "Joint example-based depth map super-resolution," in *IEEE Int. Conf. Multimedia and Expo (ICME'12)*, Jul. 2012, pp. 152–157.

[20] O. M. Aodha, N. D. F. Campbell, A. Nair, and G. J. Brostow, "Patch based synthesis for single depth image super-resolution," in *Proc. European Conf. Computer Vision (ECCV'12)*, Oct. 2012, pp. 71–84.

[21] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Magazine*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[22] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. Info. Theory*, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.

[23] M. Elad and A. M. Bruckstein, "A generalized uncertainty principle and sparse representation in pairs of bases," *IEEE Trans. Info. Theory*, vol. 48, no. 9, pp. 2558–2567, Sep. 2002.

[24] S. Hawe, M. Kleinsteuber, and K. Diepold, "Dense disparity maps from sparse disparity measurements," in *Proc. IEEE Int. Conf. Computer Vision (ICCV'11)*, Nov. 2011, pp. 2126–2133.

[25] S. Schwartz, A. Wong, and D. A. Clausi, "Saliency-guided compressive sensing approach to efficient laser range measurement," *J. Vis. Commun. Image R.*, vol. 24, no. 2, pp. 160–170, 2013.

[26] S. Schwartz, A. Wong, and D. A. Clausi, "Multi-scale saliency-guided compressive sensing approach to efficient robotic laser range measurement," in *Proc. IEEE Computer Society Conf. Computer, Robot Vision*, 2012, pp. 1–8.

[27] A. Kirmani, A. Colaco, F.N.C. Wong, and V.K. Goyal, "CODAC: A compressive depth acquisition camera framework," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process. (ICASSP'12)*, Mar. 2012, pp. 5425–5428.

[28] A. Kirmani, D. Venkatraman, D. Shin, A. Colaco, F.N. C. Wong, J.H. Shapiro, and V.K. Goyal, "First photon imaging," *Science Magazine*, vol. 343, no. 6166, pp. 58–61, Nov. 2013.

[29] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Mar. 2004.

[30] F. Chung and L. Lu, "Concentration inequalities and martingale inequalities: a survey," *Internet Mathematics*, vol. 3, no. 1, pp. 79–127, 2006.

[31] M. N. Do and M. Vetterli, "The contourlet transform: An efficient directional multiresolution image representation," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2091–2106, Dec. 2005.

[32] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, vol. 23, pp. 947–968, Apr. 2007.

[33] E. J. Candès and Y. Plan, "A probabilistic and RIPless theory of compressed sensing," *IEEE Trans. Information Theory*, vol. 57, no. 11, pp. 7235–7254, Nov. 2011.

[34] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Const. Approx.*, vol. 28, no. 3, pp. 253–263, Dec. 2008.

[35] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[36] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 53–69, Jan. 2008.

[37] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, Academic Press, Dec. 2008.

[38] D. D. Y. Po and M. N. Do, "Directional multiscale modeling of images using the contourlet transform," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1610–1620, Jun. 2006.

[39] E. J. Candès and D. L. Donoho, "Recovering edges in ill-posed inverse problems: Optimality of curvelet frames," *Annals of Statistics*, , no. 3, pp. 784–842, Aug. 2002.

[40] E. J. Candès and D. L. Donoho, "New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities," *Communications on Pure and Applied Mathematics*, vol. 57, no. 2, pp. 219–266, Feb. 2004.

[41] E. Le Pennec and S. Mallat, "Bandelet image approximation and compression," *Multiscale Model. Simul.*, vol. 4, no. 3, pp. 992–1039, 2005.

[42] M. Vetterli and J. Kovačević, *Wavelets and subband coding*, Prentice Hall, 1995.

[43] J. J. Moreau, "Proximité et dualité danes un espace hilbertien," *Bulletin de la Société Mathématique de France*, vol. 93, pp. 273–299, 1965.

[44] J. Eckstein and D. P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Math. Program.*, vol. 55, no. 3, pp. 293–318, Jun. 1992.

[45] J. Yang, Y. Zhang, and W. Yin, "An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise," *SIAM J. on Sci. Comput.*, vol. 31, no. 4, pp. 2842–2865, Jul. 2009.

[46] D. Han and X. Yuan, "A note on the alternating direction method of multipliers," *J. of Optim. Theory and Applications*, vol. 155, no. 1, pp. 227–238, Oct. 2012.

[47] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[48] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Info. Theory*, vol. 55, no. 5, pp. 2230–2249, May 2009.

[49] L. Liu, S. H. Chan, and T. Q. Nguyen, "Depth reconstruction from sparse samples: Representation, algorithm, and sampling (supplementary material)," Available online at http://arxiv.org/abs/1407.3840.

[50] S. H. Chan, T. Zickler, and Y. M. Lu, "Monte Carlo non-local means: Random sampling for large-scale image filtering," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3711–3725, Aug. 2014.

[51] F. Li, J. Yu, and J. Chai, "A hybrid camera for motion deblurring and depth map super-resolution," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'08)*, Jun. 2008, pp. 1–8.

[52] J. Park, H. Kim, Y. Tai, M.S. Brown, and I. Kweon, "High quality depth map upsampling for 3D-TOF cameras," in *Proc. IEEE Int. Conf. Computer Vision (ICCV'11)*, Nov. 2011, pp. 1623–1630.

[53] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. on Computer Vision*, vol. 47, no. 1-3, pp. 7–42, Apr. 2002.

[54] Z. Lee, J. Juang, and T. Q. Nguyen, "Local disparity estimation with three-moded cross census and advanced support weight," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1855–1864, Dec. 2013.
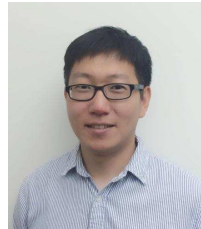
Lee-Kang Liu (S'14) received the B.S. degree in Electrical and Control Engineering from National Chiao Tung University, Hsinchu, Taiwan, in 2007, and the M.S. degree in Electrical and Computer Engineering from the University of California at San Diego, La Jolla, CA, in 2011. He is currently pursuing a Ph.D. degree in the Department of Electrical and Computer Engineering at the University of California at San Diego.

Mr. Liu worked as an intern at Cisco Inc., WebEx group on video conferencing and at Qualcomm Inc., mixed signal group, in 2012 and 2014, respectively. His research interests include image restoration, compressed sensing, 3D video processing and digital filter design.
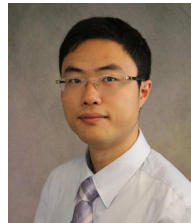
Stanley H. Chan (S'06-M'12) received the B.Eng. degree in Electrical Engineering (with first class honor) from the University of Hong Kong in 2007, the M.A. degree in Mathematics and the Ph.D. degree in Electrical Engineering from the University of California at San Diego, La Jolla, CA, in 2009 and 2011, respectively.

Dr. Chan was a postdoctoral research fellow in the School of Engineering and Applied Sciences and the Department of Statistics at Harvard University, Cambridge, MA, from January 2012 to July 2014. He joined Purdue University, West Lafayette, IN, in August 2014, where he is currently an assistant professor of Electrical and Computer Engineering, and an assistant professor of Statistics. His research interests include statistical signal processing and graph theory, with applications to imaging and network analysis. He is a recipient of the Croucher Foundation Scholarship for PhD Studies 2008-2010 and the Croucher Foundation Fellowship for Post-doctoral Research 2012-2013.

Truong Q. Nguyen (F'05) is currently a Professor at the ECE Dept., UCSD. His current research interests are 3D video processing and communications and their efficient implementation. He is the coauthor (with Prof. Gilbert Strang) of a popular textbook, Wavelets & Filter Banks, Wellesley-Cambridge Press, 1997, and the author of several matlab-based toolboxes on image compression, electrocardiogram compression and filter bank design. He has over 400 publications.

Prof. Nguyen received the IEEE Transaction in Signal Processing Paper Award (Image and Multidimensional Processing area) for the paper he co-wrote with Prof. P. P. Vaidyanathan on linear-phase perfect-reconstruction filter banks (1992). He received the NSF Career Award in 1995 and is currently the Series Editor (Digital Signal Processing) for Academic Press. He served as Associate Editor for the IEEE Transaction on Signal Processing 1994-96, for the Signal Processing Letters 2001-2003, for the IEEE Transaction on Circuits & Systems from 1996-97, 2001-2004, and for the IEEE Transaction on Image Processing from 2004-2005.