Global Auto-Regressive Depth Recovery via Iterative Non-Local Filtering

Jingyu Yang, *Senior Member, IEEE*, Xinchen Ye, *Member, IEEE*, and Pascal Frossard, *Fellow, IEEE*

***Abstract*—Existing depth sensing techniques have many shortcomings in terms of resolution, completeness, and accuracy. The performance of 3-D broadcasting systems is therefore limited by the challenges of capturing high-resolution depth data. In this paper, we present a novel framework for obtaining high-quality depth images and multi-view depth videos from simple acquisition systems. We first propose a single depth image recovery algorithm based on auto-regressive (AR) correlations. A fixed-point iteration algorithm under the global AR modeling is derived to efficiently solve the large-scale quadratic programming. Each iteration is equivalent to a nonlocal filtering process with a residue feedback. Then, we extend our framework to an AR-based multi-view depth video recovery framework, where each depth map is recovered from low-quality measurements with the help of the corresponding color image, depth maps from neighboring views, and depth maps of temporally adjacent frames. AR coefficients on nonlocal spatiotemporal neighborhoods in the algorithm are designed to improve the recovery performance. We further discuss the connections between our model and other methods like graph-based tools, and demonstrate that our algorithms enjoy the advantages of both global and local methods. Experimental results on both the Middleburry datasets and other captured datasets finally show that our method is able to improve the performances of depth images and multi-view depth videos recovery compared with state-of-the-art approaches.**

***Index Terms*—Depth recovery, multi-view depth video, auto-regressive model, nonlocal correlation, iterative filtering.**

I. Introduction

3D IMAGING applications have rapidly gained interest in recent years, as they promise to enhance the visual experience by extending the conventional 2D video to a more immersive experience. However, the technologies constituting 3D imaging systems are not fully mature yet to accomplish these targets. Depth maps need to be either captured with specialized apparatus or estimated from scene textures, and generating high-quality depth maps that are crucial in 3D computer vision stays difficulty. In short, there are two main categories of methods to obtain depth information: passive methods and active methods. Passive methods, i.e., stereo/multi-view matching [1], [2] have been an active area for several decades. However, the requirement of accurate image rectification and the inefficiency in texture-less areas have limited their practical application. Alternatively, active methods directly acquire depth information using depth sensors such as Time-of-Flight (ToF) cameras, or Microsoft Kinect. Recently, the new version of Kinect (Kinect v2), which uses the ToF technique, enables higher accuracy than the previous version Kinect v1.

While the new depth sensors are promising, the use of depth cameras is still limited by the low quality of the produced depth maps that have low resolution, noise, and depth missing in some areas. Therefore, effective post-processing and fusion techniques are needed to create high-quality depth maps for a truly 3D experience [3], [4]. Several methods have been proposed on the depth recovery from low-quality depth observations [3]–[15]. Usually, texture and depth data captured in the same view exhibit a strong structural correlation. Therefore, one can use the auxiliary texture images to enhance the low resolution depth maps by joint image filtering techniques [6], [12]–[15], global functional optimization [5], [8]–[10], or the recent deep neural networks [16], [17]. However, their performance is still subject to some artifacts, for example, jagging, blurring, and texture copying.

Moreover, the above methods mainly focus on the recovery of single depth map. Practical applications such as robotic vision and tracking require the processing of depth videos, or even multi-view depth videos. Only a few work exploit the recovery of depth video [18], [19] or multi-view depth maps [20]–[22]. However, temporal consistency enforcement by using only optical flow in these methods cannot preserve sharp depth discontinuities especially in complex texture and intensive motion areas. More investigations are still necessary for high quality depth recovery to fully consider spatial, inter-view, and inter-frame correlation.

In this paper, we present a novel framework for obtaining high-quality depth images and videos. There are two parts in our framework, i.e., AR-based single depth image recovery (ARSDIR) and multi-view depth video recovery (ARMDVR). We first derive a fixed-point iteration algorithm

Manuscript received October 1, 2017; revised January 29, 2018; accepted March 8, 2018. Date of publication April 23, 2018; date of current version March 4, 2019. This work was supported in part by the Sino Swiss Science and Technology Cooperation Program under Grant FU-06-032014 and Grant EG-08-092011, in part by the National Natural Science Foundation of China under Grant 61771339 and Grant 61702078, and in part by the Reserved Peiyang Scholar Program of Tianjin University. *(Corresponding author: Xinchen Ye.)*

J. Yang is with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China (e-mail: yjy@tju.edu.cn).

X. Ye is with the DUT-RU International School of Information Science and Engineering, Dalian University of Technology, Dalian 116100, China, and also with the Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian 116100, China (e-mail: yexch@dlut.edu.cn).

P. Frossard is with the Signal Processing Laboratory 4, École Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland (e-mail: pascal.frossard@epfl.ch).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TBC.2018.2818405

0018-9316 © 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

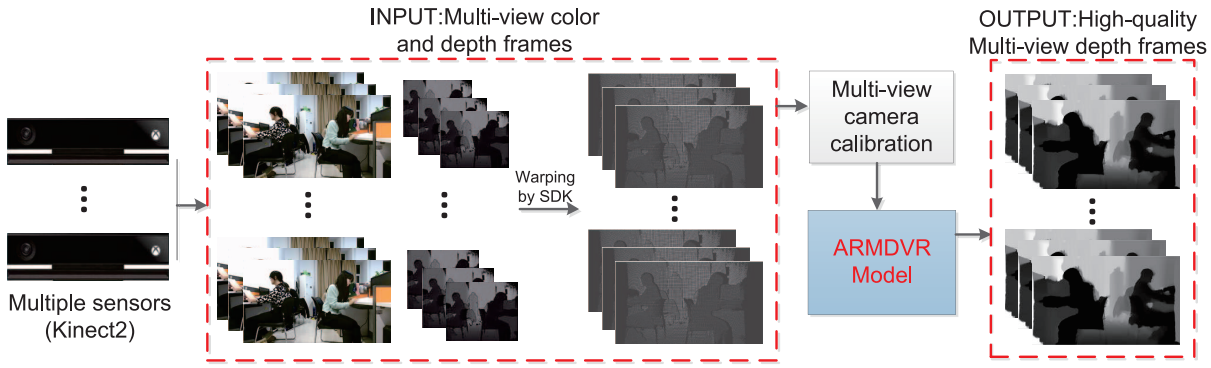


Fig. 1. The overall framework of multi-view depth video recovery. The built-in calibration applied in the Kinect v2 and multi-view sensor calibration are both used to correct lens distortion and warp the depth map from different views to the current view.

under the basic AR model to recover single depth images. Each iteration is equivalent to a non-local filtering process on the observed depth map and the recovered one at the last iteration with a residue feedback. Then, we extend the original spatial AR model to multi-view depth video settings by exploiting spatial, inter-view, and inter-frame correlations. The AR coefficients on nonlocal spatiotemporal neighborhoods in our algorithms are carefully designed, and are adaptively updated on-the-fly for better performance. Then, we discuss the connections and differences between our model and other closely-related methods, including local filtering based methods, global optimization based methods, and graph-based methods.

We construct a multi-view depth sensing prototype using Kinect v2 (shown in Fig. 1) to evaluate the recovery performance in multi-view settings. Experimental results on both the Middleburry datasets and captured real datasets finally show that our method is able to improve the performances of depth images and multi-view depth videos recovery over several state-of-the-art approaches, as our algorithms own the advantages of both global and local filtering methods.

The contributions of our work are summarized into the following three aspects:

1) A fixed-point iteration algorithm with updating AR coefficients on nonlocal spatiotemporal neighborhoods is derived under global AR modeling. The proposed algorithm is applicable to the whole depth maps, remedying the difficulty of the direct least squares approach [10] in inverting large matrices, and shortens the running time compared to classical methods [8]–[10]. The ability to refresh parameters on-the-fly also improves recovery performance over the direct least squares solution.

2) A new model is proposed for depth recovery on multi-view depth videos by exploiting spatial, inter-view, and inter-frame correlations, while former schemes [9], [10], [18] only consider part of this information. This work also provides in-depth discussions on the connections and differences between our model and other depth recovery methods including graph-based techniques.

3) Comprehensive evaluation on public datasets with three synthetic degradation types (undersampling, ToF-like degradation, and Kinect-like degradation) compared with state-of-the-art methods. This work also setups a multi-view depth sensing prototype, which evaluates the proposed multi-view depth video recovery algorithm in scenarios close to practical settings.

The paper is organized as follows. In Section II, we present a brief overview of related work. We describe our ARSDIR algorithm and ARMDVR algorithm in Section III and Section IV, respectively, and discuss the connections between our proposed method and other recovery methods in Section V. Finally, experimental results and conclusion are shown in Section VI and Section VII, respectively.

II. Related Work

The depth recovery task is to reconstruct high quality depth information and fill the missing depth values from lower resolution observations. The depth information and texture information are two descriptions of the same scene, which however present strong structural correlations. After alignment by warping, the structural correlation between the depth map and the texture image can be exploited for information recovery. In this section, we briefly review the recent work related to the depth recovery task.

A. Depth Image Recovery

Depth image recovery can be classified in two categories, namely the global methods and local methods.

1) Global Methods: The global methods recast the depth recovery task as a global optimization problem, which consists in a data term and a smooth term in an energy function. The data term penalizes the difference between the observation depth values and the recovered ones, while the smooth term penalizes the difference between neighboring pixels on the high-resolution depth map.

Diebel and Thrun [8] have proposed a two-layer MRF model to represent the correlation between range measurements and solved the MRF optimization with a conjugate gradient algorithm. This method is able to improve the quality of depth maps, but tends to over-smooth the depth images. To reduce over-smoothing, Hannemann *et al.* [23] have incorporated the amplitude values generated by Time-of-Flight camera into an

MRF model to improve the quality of interpolation. The amplitude can be evaluated as a confidence measurement for the depth values. Lu *et al.* [24] have further extended this work by designing a data term that fits the characteristics of depth maps. Huhle *et al.* [25] have added a third layer to the MRF framework [8], where image gradients are encoded as nodes in the graph. Zhu *et al.* [26] have fused the stereo parallax and depth information into a unified MRF model, thus improving the accuracy and robustness of depth recovery. Park *et al.* [9] have added a weighting scheme to the smooth term, which involves edge, gradient, and segmentation information extracted from high quality color images, and used an extra non-local term to regularize depth maps. Ferstl *et al.* [27] have modeled the smooth term as a second order total generalized variation regularization, and guided the depth upsampling with an anisotropic diffusion tensor calculated from a high-resolution intensity image. Yang *et al.* [10], [28] have proposed the adaptive color-guided auto-regressive models for high quality depth recovery. The depth recovery task is formulated as the minimization of AR prediction errors subject to measurement consistency using least squares. Li *et al.* [29] proposed a hierarchical global optimization framework based on weighted least squares (WLS) technique. In the MRF framework, Zuo *et al.* [30] explicitly model the discontinuity inconsistency between the depth map and color image in the smoothness term to reduce texture copy and blurring artifacts.

These global models could ensure the achievable mathematical optimality when using differentiable metrics, and a number of well-understood numerical algorithms such as belief propagation, graph cut, and least squares. However, such algorithms require intensive computation to solve large-scale problems, and some are even not applicable due to the difficulty to inverting very large matrices. Moreover, model parameters determined from the degraded depth map are sub-optimal. Pizarro *et al.* [31] proposed an iterative procedure to minimize the convex nonlocal data and smoothness model. Inspired by this, in this paper, we propose a new fixed-point iteration algorithm for the AR modeling, thus remedying the difficulty of the direct least squares approach [10] in inverting large matrices. The weighting scheme is updated on-the-fly to make the proposed framework more adaptive to depth content.

2) Local Methods: The local methods for depth recovery generally use local filters such as bilateral filters and non-local means (NLM) filters [32]–[34]. The depth value at a given pixel is refreshed by a weighted average of depth values from neighboring pixels, and the weights are predicted by some weighting strategies derived from the color image. Joint bilateral filtering [35] has been also used in depth recovery algorithms using high quality auxiliary color images [32], [33], [36]. Yang *et al.* used the joint bilateral filtering on cost values [32] to super-resolve range images and proposed a hierarchical joint bilateral filtering scheme [37] for depth map upsampling. To fully exploit correlation across more domains, the bilateral filter is extended into multi-lateral cases [4], [38], [39]. Min *et al.* [18] have proposed a weighted mode filtering method based on a joint histogram of depth image, where the final solution is determined by seeking a global mode on the histogram. He *et al.* [40] have investigated guided filtering to derive an edge-preserving smoothing operator like the popular bilateral filter. Lu *et al.* [41] have formulated the filtering process as a local multipoint regression problem, consisting of multipoint estimation within a shape-adaptive local support, and aggregation of a number of multipoint estimates available for each point. It models a zero-order or linear relation between observed low resolution depth patch and color patch. Liu *et al.* [11] have used a geodesic distance to compute the filtering coefficients based on the similarity between pixels. Barron and Poole [42] have proposed an edge-aware smoothing method based on fast bilateral solver that combines the flexibility and speed of simple filtering approaches.

In general, depth recovery schemes based on filtering techniques enjoy the simplicity in design and implementation, lower computational complexities, and often good recovery results. However, the short-sighted local judgement cannot provide enough information to recover the global structure, and may introduce annoying artifacts in regions where the associated color image contains rich textures. In this paper, we propose the iterative non-local filtering algorithm to tackle the global AR model, which enjoy both the merits of global methods and local methods.

B. Depth Recovery for Depth Videos

Beyond the frame-wise recovery using single depth map based methods, the recovery of depth videos usually involves mechanisms that exploit temporal correlation (or even view correlation in multi-view settings). Duan *et al.* [43] have taken into account the temporal and spatial factors when using the graph cut based on epipolar rectification. The penalty function with coherence factor is introduced for temporal consistency. Min *et al.* [18] have also extended their spatial method for temporally neighboring frames. Simple optical flow estimation and patch similarity measure are used for obtaining the high-quality depth video in an efficient manner. Sheng *et al.* [19] have proposed an intrinsic static structure, which defines a static structure for the captured scene to enhance the depth video. The structure is estimated iteratively by a probabilistic generative model with sequentially incoming depth frames. Wang *et al.* [44] uses nonlocal regression and total variation prior in the global function to upsample multi-view RGB-D images simultaneously. Zhang *et al.* [45] have proposed a unified scheme for texture super-resolution and depth estimation from binocular video. Alternatively, Kim *et al.* [20] have performed depth balancing and multi-view depth fusion based on a confidence map in order to enhance multi-view depth maps obtained from multiple ToF sensors. Liu *et al.* [21] have proposed a gradient-domain based enhancement method for multi-view depth. The method exploits the coherence of both temporal and inter-view dimensions in addition of the spatial one. Finally, Choi *et al.* [22] have improved the quality of the depth map corresponding to each color view by increasing its spatial resolution and enforcing interview coherence. However, these methods only use part of the correlations among spatial, inter-view and inter-frame information. Besides, temporal consistency enforcement by using only optical flow in these

methods cannot preserve sharp depth discontinuities especially in the motion areas.

Based on the basic AR model, we propose a new model on multi-view depth videos by exploiting spatial, inter-view, and inter-frame correlations, and derive the corresponding fixed-point iteration algorithm to solve the model.

III. AR-Based Single Depth Image Recovery (ARSDIR)

In this section, we propose an effective algorithm for single depth image recovery (ARSDIR), using an auto-regressive (AR) model. The AR model has shown to be an effective signal model in recovering high quality depth maps from degraded ones [10], but recovery algorithms have so far hit limitation in terms of computational complexity. We show here a computationally effective algorithm based on fixed-point iteration method for recovering static depth maps.

A. Motivation

It has been shown that the AR model describes well the depth maps that mainly containing smooth regions separated by curves [10]. Denote by $\widetilde{\boldsymbol{D}}$ the observed depth map, and $\boldsymbol{C}$ the corresponding color image. Let $\boldsymbol{D}$ be the depth map to be recovered, which has the same size as the color image $\boldsymbol{C}$. Denoting by p the pixel index, the depth (color) value at p is represented by $\boldsymbol{D}_p$ ($\boldsymbol{C}_p$). The depth recovery problem based on the AR model is written as follows:

$$\min_{\boldsymbol{D}} \underbrace{\sum_p h_p\big(\boldsymbol{D}_p - \widetilde{\boldsymbol{D}}_p\big)^2}_{\boldsymbol{E}_{\text{data}}} + \lambda \underbrace{\sum_p \left(\boldsymbol{D}_p - \sum_{q\in\mathcal{N}(p)\backslash p} w_{p,q}\boldsymbol{D}_q\right)^2}_{\boldsymbol{E}_{\text{AR}}}, \tag{1}$$

where $\boldsymbol{E}_{\text{data}}$ is the data term to make the recovered depth consistent with the observation, $\boldsymbol{E}_{\text{AR}}$ is the AR term to impose AR regularization on the recovered depth map. $\mathcal{N}(p)$ is the neighborhood of pixel p, and the parameter λ represents the weight between the two terms in (1). The parameter h_p can be 0 or 1, and describes whether $\widetilde{\mathbf{D}}$ appears as a valid observation at p (1 for valid pixel, and 0 otherwise). Finally, the weight $w_{p,q}$ denotes the AR coefficient for the pixel q in the *deleted neighborhood* $\mathcal{N}(p)\backslash p$ of p, which is defined according to both the local correlation in the initial depth map and the nonlocal similarity in the accompanied high quality color image:

$$w_{p,q} = \exp\left(-\frac{\big(\widetilde{\boldsymbol{D}}_p - \widetilde{\boldsymbol{D}}_q\big)^2}{\sigma_1^2}\right)\exp\left(-\frac{\big\|\boldsymbol{B}_p \circ \big(\mathcal{P}_p - \mathcal{P}_q\big)\big\|_2^2}{\sigma_2^2}\right), \tag{2}$$

where σ_1 and σ_2 are the decay rate of the range filter and color filter, respectively. Here, $\mathcal{P}_p$ denotes an extracted patch centered at p in color image, "$\circ$" represents the element-wise multiplication. $\boldsymbol{B}_p$ represents the bilateral filter kernel around pixel p on the color image.

The above AR function (1) can be written in a matrix form and recast as a least square problem as:

$$\min_{\boldsymbol{d}} \big\|\widetilde{\boldsymbol{d}} - \boldsymbol{H}\boldsymbol{d}\big\|_2^2 + \lambda\|\boldsymbol{d} - \boldsymbol{Q}\boldsymbol{d}\|_2^2, \tag{3}$$

where $\boldsymbol{H}$ and $\boldsymbol{Q}$ are the observation matrix (constructed by $\boldsymbol{h}_p$) and AR coefficient matrix, respectively; $\widetilde{\boldsymbol{d}}$ and $\boldsymbol{d}$ are the vector forms of $\widetilde{\mathbf{D}}$ and $\boldsymbol{D}$, respectively. The solution of problem (3) can be obtained by solving the normal equation:

$$\underbrace{\Big(\boldsymbol{P}^\top\boldsymbol{P} + \lambda(\boldsymbol{I}-\boldsymbol{Q})^\top(\boldsymbol{I}-\boldsymbol{Q})\Big)}_{\boldsymbol{A}}\boldsymbol{d} = \underbrace{\boldsymbol{P}^\top\widetilde{\boldsymbol{d}}}_{\boldsymbol{b}}, \tag{4}$$

Besides the direct inversion approach, there are a number of alternative algorithms that are more efficient in terms of computation or memory, including gradient-type algorithms (with various variants and step-size strategies) and quasi-Newton algorithms such as BFGS and L-BFGS [46]. However, the numerical stability of these algorithms largely depends on the invertibility of $\boldsymbol{A}$. The first term $\boldsymbol{P}^\top\boldsymbol{P}$ in $\boldsymbol{A}$ is a diagonal sampling matrix, and is ill-conditioned. Besides, the invertibility of $\boldsymbol{Q}^\top\boldsymbol{Q}$ is largely determined by the effective support of the prediction candidates for each target pixel, and $\boldsymbol{Q}^\top\boldsymbol{Q}$ becomes ill-conditioned when a few pixels have not enough effective prediction candidates, especially in the region of rich textures. Combining $\boldsymbol{P}$ and $\boldsymbol{Q}$, the resulting matrix $\boldsymbol{A}$ might be still highly ill-conditioned. Any small change in $\boldsymbol{A}$ can cause a large change in the variable $\boldsymbol{d}$, which would severely degrade the performance of the depth recovery process.

Moreover, the scale of the matrix $\boldsymbol{A}$ is the square of the depth-map size, which is beyond the capability of current desktop computer to recover the whole depth image at once due to limited memory. The depth image should be divided into overlapping patches, and the recovery process is done on each patch to cope with the memory limitations. Then, the recovered patches are integrated into an entire image. This patch division and re-integration process would yet increase the amount of calculation, and inverting a large number of matrices (though with a smaller sizes) is still quite time-consuming. Although the AR model itself has interesting properties, the deficiency of the above algorithm itself impedes its actual performance. This motivates us to find a better solution for the global AR modeling.

B. Fixed-Point Iteration Algorithm

We now derive a robust and stable iterative procedure to efficiently minimize the energy function in problem (1). The first order optimality conditions imply that the partial derivative with respect each component $\boldsymbol{D}_p$ equals to zero:

$$\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{D}_p} = 0, \tag{5}$$

where the partial derivative $\partial\boldsymbol{E}/\partial\boldsymbol{D}_p$ is calculated as follows.

$$\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{D}_p} = \frac{\partial \boldsymbol{E}_{\text{data}}}{\partial \boldsymbol{D}_p} + \lambda\frac{\partial \boldsymbol{E}_{\text{AR}}}{\partial \boldsymbol{D}_p},$$

$$\frac{\partial \boldsymbol{E}_{\text{data}}}{\partial \boldsymbol{D}_p} = 2\boldsymbol{h}_p\big(\boldsymbol{D}_p - \widetilde{\boldsymbol{D}}_p\big),$$

$$\frac{\partial \boldsymbol{E}_{\mathrm{AR}}}{\partial \boldsymbol{D}_p} = 2\Bigg(\boldsymbol{D}_p - \sum_q w_{p,q}\boldsymbol{D}_q - \sum_r w_{r,p}\Bigg(\boldsymbol{D}_r - \sum_q w_{r,q}\boldsymbol{D}_q\Bigg)\Bigg), \quad (6)$$

We organize Eq. (5) and Eq. (6) into the following fixed point form

$$\boldsymbol{D}_p = \Bigg(\boldsymbol{h}_p\widetilde{\boldsymbol{D}}_p + \lambda\sum_q w_{p,q}\boldsymbol{D}_q + \lambda\sum_r w_{r,p}\Bigg(\boldsymbol{D}_r - \sum_q w_{r,q}\boldsymbol{D}_q\Bigg)\Bigg)\Big/\big(\boldsymbol{h}_p + \lambda\big). \quad (7)$$

Denote by k the iteration index, Formula (7) is readily written into the following fixed-point iteration:

$$\boldsymbol{D}_p^{(k)} = \Bigg(\boldsymbol{h}_p\widetilde{\boldsymbol{D}}_p + \lambda\sum_q w_{p,q}\boldsymbol{D}_q^{(k-1)} + \lambda\sum_r w_{r,p}\Bigg(\boldsymbol{D}_r^{(k-1)} - \sum_q w_{r,q}\boldsymbol{D}_q^{(k-1)}\Bigg)\Bigg)\Big/\big(\boldsymbol{h}_p + \lambda\big), \quad (8)$$

where the depth map is initialized as the observed depth samples, i.e., $\boldsymbol{D}^{(0)} = \widetilde{\boldsymbol{D}}$. The iterative procedure approaches the solution of the first order optimality conditions of Eq. (5), and hence the one of the AR-based depth recovery model in Eq. (1) [46]. The above iterative formula is essentially a filtering processing of the observed depth map $\widetilde{\boldsymbol{D}}$ and the recovered one $\boldsymbol{D}^{(k)}$ by the last iteration with a residue feedback. Concretely, the first term $\widetilde{\boldsymbol{D}}_p$ re-includes the observed samples to ensure the consistency with the measurements; the second term $\sum_q w_{p,q}\boldsymbol{D}_q$ is a standard linear filtering on $\boldsymbol{D}^{(k)}$ with AR prediction coefficients; the third term $\sum_r w_{r,p}(\boldsymbol{D}_r - \sum_q w_{r,q}\boldsymbol{D}_q)$ is a residue feedback that compensates for the AR prediction errors to the filtered depth map; finally, the denominator $\boldsymbol{h}_p + \lambda$ is a normalization term for stable filtering. The convergence is determined by the relative error $||\boldsymbol{D}^{(k)} - \boldsymbol{D}^{(k-1)}||_2/||\boldsymbol{D}^{(k-1)}||_2$. Empirically, the algorithm achieves stable recovery results after several iterations (see Fig. 3), and therefore we set a maximal number of iterations in our implementation.

The corresponding algorithm procedures are summarized in Algorithm 1.

Based on the above derivation, the global optimization of Eq. (3) is broken down into a sequence of iterative fixed-point filtering steps, which does not require constructing and inverting the large-scale matrix $\boldsymbol{A}$ in Eq. (4). Unlike well-known iterative algorithms such as the conjugated gradient type algorithms and BFGS-type algorithms, each iteration could be efficiently implemented in-place, hence without patch division and re-integration process. Moreover, the parameters in our algorithm are well interpreted from the perspective of image filtering, and could be updated according to the recovered depth map at each iteration to achieve better performance, which is demonstrated in Section VI.

Algorithm 1 The Fixed-Point Iteration Algorithm for ARSDIR

Input: $\boldsymbol{D}^{(0)} = \widetilde{\boldsymbol{D}}$: initial depth map; $\boldsymbol{C}$: the corresponding color image; ϵ: the stopping relative error; K: the maximal number of iterations.

```
while ||D^(k) − D^(k−1)||_2/||D^(k−1)||_2 > ε || k < K do
  for each pixel p in D^(k) do
    Update AR coefficients w_{p,q} using Eq. (2);
    Estimate D_p^(k) via nonlocal filtering with Eq. (8);
  end for
  k ← k + 1;
end while
```

IV. AR-Based Multi-View Depth Video Recovery (ARMDVR)

We now extend the baseline ARSDIR framework to more general multi-view/multi-frame settings by incorporating inter-frame and inter-view correlations for improved performance. Correspondingly, we derive a new fixed-point iteration algorithm and present the design of spatiotemporal neighborhoods and the corresponding adaptation of the AR weights.

A. The ARMDVR Model

In multi-view settings, the current view can have multiple depth map candidates by warping depth samples from neighbor views. The depth maps warped from other views can provide depth information in occluded areas of the current view. Denote by $\widetilde{\boldsymbol{D}}^{[i]}$ the depth map captured by the $\boldsymbol{i}^{\text{th}}$ camera and registered to the current view. To avoid heavy notations, the depth map at the current view to be recovered is denoted by $\boldsymbol{D}$ without view index. We propose the following model for multi-view depth image recovery from multiple observed depth samples:

$$\min_{\boldsymbol{D}} \sum_i \sum_p h_p^{[i]}\Big(\boldsymbol{D}_p - \widetilde{\boldsymbol{D}}_p^{[i]}\Big)^2 + \lambda\sum_p\Bigg(\boldsymbol{D}_p - \sum_{q\in\mathcal{N}(p)\backslash p} w_{p,q}\boldsymbol{D}_q\Bigg)^2, \quad (9)$$

The above model consists of multiple data terms that measure the proximity of $\boldsymbol{D}$ to the measured depth maps, and the AR regularization term that measures the similarity between the depth value at the current pixel p and its neighboring pixels in $\mathcal{N}(p)\backslash p$. Differently from Eq. (1) in which h_p indicates whether the depth map has a valid observed sample at the pixel p, $h_p^{[i]}$ represents here the weight of the depth value $\widetilde{\boldsymbol{D}}_p^{[i]}$ from the i^{th} initial depth map. The reliability of observed depth samples from neighboring views mainly depends on calibration accuracy and rounding errors in view warping. The calibration accuracy is mainly affected by the system

setup: the larger the baseline distance between two cameras is, the larger calibration errors would be. The rounding errors could be suppressed by using advance interpolation methods, but we observe that the improvement is marginal in our framework. Hence, the reliability $h^{[i]} \in [0, 1]$ is designed to be zero for unobserved pixels, and to be monotonically decreasing with respect to the baseline distance for available pixels:

$$h_p^{[i]} = \begin{cases} 0, & \widetilde{\boldsymbol{D}}_p^{(i)} = 0, \\ \exp(-b_i/\sigma), & \text{otherwise}, \end{cases} \tag{10}$$

where b_i is the baseline distance between the i^{th} camera and the reference view, and σ controls the decreasing rate. If there is no depth sample in $\widetilde{\boldsymbol{D}}^{[i]}$, $h_p^{[i]}$ is equal to zero. When b_i is 0, i.e., $\widetilde{\boldsymbol{D}}^{[i]}$ represents the depth map of the current view, $h_p^{[i]}$ is set to one.

There would be flickering artifacts if the above model (9) is applied to multi-view video sequences in the frame-by-frame manner. We want to ensure consistency across time and therefore incorporate temporal correlation into the AR-based depth recovery framework. For clear notations, we introduce a pair of superscripts $[i, t]$ to denote the view index and frame index, respectively. For example, $\boldsymbol{D}^{[i,t]}$ is the t^{th} depth frame at the i^{th} view. Without loss of generality, we denote the current depth frame at the current view to be recovered by $\boldsymbol{D}^{[0,0]}$, and by $\boldsymbol{D}$ for compact presentation omitting the superscripts. Our final ARMDVR model reads:

$$\begin{aligned} \min_{\boldsymbol{D}} \sum_{i\in\Gamma_{\text{view}}} \sum_p h_p^{[i,0]} \left(\boldsymbol{D}_p - \widetilde{\boldsymbol{D}}_p^{[i,0]}\right)^2 \\ + \lambda \sum_p \left(\boldsymbol{D}_p - \sum_{t\in\Gamma_{\text{temp}}} w_{p,p_t} \sum_{q_t\in\mathcal{N}(p_t)\backslash p_t} w_{p_t,q_t} \boldsymbol{D}_{q_t}^{[0,t]} \right)^2 \end{aligned} \tag{11}$$

where Γ_{view} and Γ_{temp} are the index sets of involved neighboring views and neighboring frames. The pixel p_t represents a corresponding pixel of p in the t^{th} frame, and q_t is within the deleted neighborhood of p_t, denoted by $\mathcal{N}(p_t)\backslash p$. $\boldsymbol{D}_{q_t}^{[i,t]}$ is the depth value at q_t in the t^{th} depth frame $\boldsymbol{D}^{[i,t]}$. The weight w_{p,p_t} is the temporal AR weight determined by the similarity between the patch around p and that around p_t, and w_{p_t,q_t} is the spatial AR weight determined by the similarity between pixel p_t and pixel q_t in the t^{th} frame.

Fully determining model (11) involves two types of correspondence: 1) inter-view correspondence and 2) inter-frame correspondence. The data term incorporates observed samples from neighboring views, where corresponding inter-view pixels are established by view warping with calibration parameters. In the AR term, AR coefficients are calculated on 3D spatiotemporal neighborhoods, where the temporal correspondence is established by optical flow as detailed in Section (IV-B).

Similarly to the Algorithm 1 for the ARSDIR model of Eq. (1), we also employ a fixed-point iteration algorithm to solve the problem in Eq. (11). It is also derived by the first order optimality conditions, yielding the following iterative procedure:

$$\begin{aligned} \boldsymbol{D}_p^{(k+1)} = \sum_i h_p^{[i,0]} \widetilde{\boldsymbol{D}}_p^{[i,0]} + \lambda \sum_t w_{p,p_t} \left(\sum_{q_t} w_{p_t,q_t} \boldsymbol{D}_{q_t}^{[0,t](k)} \right. \\ \left. + \sum_{r_t} w_{r_t,p_t} \left(\boldsymbol{D}_{r_t}^{[0,t](k)} - \sum_{q_t} w_{r_t,q_t} \boldsymbol{D}_{q_t}^{[0,t](k)} \right) \right) \\ \Big/ \left(\sum_i h_p^{[i,0]} + \lambda \right) \end{aligned} \tag{12}$$

where k is the iteration index and r_t is the pixel index belonging to the deleted neighborhood around p_t, i.e., $\mathcal{N}(p_t)\backslash p_t$. The depth map is initialized as $\boldsymbol{D}_p^{(0)} = \sum_i h_p^{[i,0]} \widetilde{\boldsymbol{D}}_p^{[i,0]}$, which fuses the depth maps $\widetilde{\boldsymbol{D}}_p^{[i,0]}$ warped from neighbouring views. Then, the depth map is recovered via iterative filtering on the initial depth map and the recovered depth map at the last iteration. The overall algorithm to solve the ARMDVR model is summarized in Algorithm 2.

Algorithm 2 ARMDVR Algorithm

Input: $\boldsymbol{D}$: depth frame to be recovered; $\boldsymbol{D}^{[i,t]}$: neighboring frame of $\boldsymbol{D}$; $\boldsymbol{C}$: the current color frame; $\boldsymbol{C}^{[i,t]}$: the neighboring i^{th} color frame at t^{th} view;
while note converged **do**
  for each pixel p in the current frame $\boldsymbol{D}^{(k)}$ **do**
    Compute the spatial weighting w_{p_t,q_t} using $\boldsymbol{C}^{[0,t]}$ and $\boldsymbol{D}^{[0,t](k-1)}$;
    Compute the temporal weighting w_{p,p_t} between $\boldsymbol{C}$ and $\boldsymbol{C}^{[0,t]}$;
    Estimate $\boldsymbol{D}_p^{(k)}$ by the filtering in Formula (12);
  end for
  $k \leftarrow k + 1$;
end while

B. AR Coefficients on Nonlocal Spatiotemporal Neighborhoods

It is straightforward to adapt the AR weighting scheme (2) for the depth video recovery model by extending 2D patches to 3D patches and searching nonlocal similar 3D patches in the video volume. However, such a 3D nonlocal search requires huge amount of computation. To limit the computational complexity, we decouple the spatial dimension and the temporal dimension, and operate on 2D patches in the formation of spatiotemporal neighborhoods.

When enforcing temporal consistency, the information of the temporal neighbors should be incorporated in a way that is robust to errors on the depth discontinuities. As illustrated in Fig. 2, we first establish rough pixel-wise temporal correspondences of the current color frame, to neighboring frames with optical flow [47]. Denote by $\tilde{p}_t$ the rough correspondence of pixel p in the neighboring frame with index t. Let $\mathcal{P}_p$ be a patch centered at p in the color frame. Then, we find the most similar patch $\mathcal{P}_{p_t}$ around the roughly estimated patch $\mathcal{P}_{\tilde{p}_t}$ in a searching window $\mathcal{N}(\tilde{p}_t)$ by using the approximate K-nearest

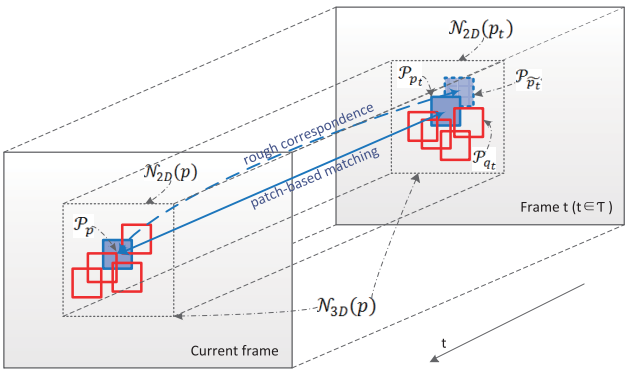


Fig. 2. Illustration of nonlocal spatiotemporal neighbourhood between two consecutive frames for the calculation of AR coefficients. The similarity between two pixels are estimated from the associated patches around.

neighbors structure (AKNN) [48],[1] where the matching cost is measured by the sum of square distances (SSD):

$$p_t = \arg \min_{q_t \in \mathcal{N}(\tilde{p}_t)} \left\| \mathcal{P}_p - \mathcal{P}_{q_t} \right\|_2 \tag{13}$$

where $\mathcal{N}(\tilde{p}_t)$ is the neighbourhood around $\tilde{p}_t$ in the t^{th} frame.

Then, the spatiotemporal neighbourhood, denoted by $\mathcal{N}_{3D}(p)$, is formed by stacking the 2D neighbourhoods of all the temporal correspondences of p, i.e., $\mathcal{N}_{3D}(p) = \{\mathcal{N}_{2D}(p_t) | t \in \Gamma_{\text{temp}}\}$. Fig. 2 shows the construction details of the spatiotemporal neighbourhood at pixel p between two consecutive frames. The AR coefficient on the spatiotemporal neighbourhood is then defined as

$$w_{p,q_t} = \frac{1}{c} w_{p,p_t} w_{p_t,q_t}, \tag{14}$$

where w_{p,p_t} measures the correlation between pixel p and its temporal correspondence p_t, w_{p_t,q_t} measures the correlation between pixel p_t and pixel q_t in frame at time t, and $c := \sum_{q^t} w_{p,q^t}$, $q^t \in \mathcal{N}_{3D}(p)$, is the normalization factor. The weight w_{p_t,q_t} is calculated according to Eq. (2), and w_{p,p_t} is calculated as the similarity between two patches around p and p_t:

$$w_{p,p_t} = \exp\left(-\frac{\left\| \mathcal{P}_p - \mathcal{P}_{q_t} \right\|_2^2}{\sigma_3^2} \right), \tag{15}$$

where σ_3 controls the decay rate of the similarity.

In the global optimization scheme of Eq. (4), the AR coefficients are set only once before solving the normal equation. One could also update the AR coefficients using the recovered depth image, and then perform global optimization for better depth recovery. However, this would require significant amounts of computation. One merit of the proposed iterative scheme is that AR coefficients can be updated on-the-fly using the intermediate recovered depth map $\boldsymbol{D}^{(k)}$. Concretely, the AR coefficients are initialized according to Eq. (2) for depth image recovery and to Eq. (14) for depth video recovery with the assistance of color images. As iterations go on, AR coefficients are updated using the intermediate depth map $\boldsymbol{D}^{(k)}$ and the associated color image. Since the recovered depth map $\boldsymbol{D}^{(k)}$ becomes more reliable than the observed depth map $\boldsymbol{D}^{(0)}$, we simplify the patch-based color similarity measurement $\|\boldsymbol{B}_p \circ (\mathcal{P}_p - \mathcal{P}_q)\|_2^2$ in Eq. (2) into the pixel-based one $\|\boldsymbol{C}_p - \boldsymbol{C}_q\|_2^2$, where $\boldsymbol{C}_p$ and $\boldsymbol{C}_q$ are color values at p and q. This simplification also reduces the required computation in patch-wise similarity calculation. To sum up, the iterative updating strategy does not only improve depth recovery performance, but also lowers the computational complexity.

[1]Note that AKNN structure aims to find non-local patch similarities using fast searching strategy.

V. Connections to Other Methods

This section discusses the connections between our proposed method and other closely-related methods, including local filtering based methods, global optimization methods, and graph model based methods.

A. Local Filtering Based Methods

The basic form of local filtering is essentially the weighted average of neighbouring samples:

$$\boldsymbol{D}_p = \sum_{q \in \mathcal{N}(p) \setminus p} w_{p,q} \boldsymbol{D}_q, \tag{16}$$

where the weights are usually normalized, i.e., $\sum_q w_{p,q} = 1$. The filter in Eq. (16) is also called *shrinking smoother* [49]. Most local filters such as Gaussian filter, bilateral filter and its variants [50], and nonlocal means filter [51] fall into this framework, differing in the determination of weights $\{w_{p,q}\}$. For example, bilateral filter introduces a range filter to consider the similarity of signal intensities. Nonlocal means filters further exploits the signal similarity at the patch rather than pixel level.

Comparing our proposed method in Eq. (8) with Eq. (16), each filtering iteration contains the shrinking smoother as a special case, in which no initial depth map and residue feedback is involved. An iterative shrinking smoother would yield stable smoothing results (usually over-smoothed) [49], while the results from our algorithm are consistent with the observations thanks to the inclusion of the observed depth samples at each iteration. Regarding to the residue feedback $\sum_r w_{r,p}(\boldsymbol{D}_r^{(k-1)} - \sum_q w_{r,q} \boldsymbol{D}_q^{(k-1)})$, the AR prediction errors at pixels whose neighbourhood containing pixel p are weighted averages, and then added back to the filtering value as a compensation. This feedback mechanism makes our algorithm more robust than other filtering methods, particularly in the prediction of edges and fine structures (see more details in Section VI).

B. Global Optimization Based Methods

Global optimization-based depth recovery methods assume local smoothness, which are usually modeled by the Markov random field (MRF). Using the same notations as our model, the MRF-based global model in [7]–[9], [27] and [29] can be unified into the following formulation:

$$\min \sum_p h_p \left(\boldsymbol{D}_p - \widetilde{\boldsymbol{D}}_p \right)^2 + \lambda \sum_p \sum_{q \in \mathcal{N}(p) \setminus p} w_{p,q} \left(\boldsymbol{D}_p - \boldsymbol{D}_q \right)^2, \tag{17}$$

where different weighting schemes $\{w_{p,q}\}$ are designed in particular methods. For example, Park *et al.* [9] used

edge, gradient, and segmentation information extracted from high quality color images to construct the weights, while Ferstl *et al.* [27] introduced an anisotropic diffusion tensor calculated from the color image. We can also derive a fixed-point algorithm [7], [29] from the global model (17):

$$\boldsymbol{D}_p = \frac{h_p \widetilde{\boldsymbol{D}_p} + \lambda \sum_q w_{p,q} \boldsymbol{D}_q}{h_p + \lambda}, \tag{18}$$

Comparing Eq. (18) and Eq. (8), our AR-based iterative algorithm is an augmented version of the MRF-based iterative algorithm by adding a feedback of AR prediction residue $\sum_r w_{r,p}(\boldsymbol{D}_r - \sum_q w_{r,q}\boldsymbol{D}_q)$, which contributes to the performance improvement as shown in Section VI.

C. Graph-Model Perspective of AR Modeling

If we denote by $\boldsymbol{K}(p,q) = w_{p,q}$ the similarity between pixels p and q, the matrix form of the filtering in Eq. (8) on the whole depth image is written as:

$$\boldsymbol{d^*} = \boldsymbol{N}^{-1}\boldsymbol{K}\boldsymbol{d}, \tag{19}$$

where $\boldsymbol{d^*}$ and $\boldsymbol{d}$ represents the vector form of the filtered result and the observed depth map, respectively. $\boldsymbol{N}$ is a diagonal matrix in which each row is normalized by $\sum_q \boldsymbol{K}(p,q)$.

Note that $\boldsymbol{K}$ also provides an interpretation of the structure of the latent depth image as a graph. Consider the depth image as a graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$, where $\boldsymbol{V}$ is the set of all pixels in the image and $\boldsymbol{E}$ represents the edges that link pairwise neighbouring pixels. Let $\boldsymbol{K}(p,q)$ be the edge weight between p and q. Now, we define the Laplacian of the graph. Let n_p be the degree of vertex p, which is equal to the sum of the weights on the edges connected to this vertex. For a graph with an adjacent matrix $\boldsymbol{K}$, we get $n_p = \sum_q \boldsymbol{K}(p,q)$. With the degree matrix $\boldsymbol{N}(p,p) = n_p$, the graph Laplacian is given by

$$\mathcal{L} = \boldsymbol{N}^{-1}(\boldsymbol{N} - \boldsymbol{K}) = \boldsymbol{I} - \boldsymbol{N}^{-1}\boldsymbol{K} = \boldsymbol{I} - \boldsymbol{W}. \tag{20}$$

Ideally, the filtering of a ground-truth depth image should be the same as the input, i.e., $\boldsymbol{d^*} = \boldsymbol{W}\boldsymbol{d^*}$, which is the assumption behind the AR model [10]. We can therefore obtain:

$$\boldsymbol{d^*} = \boldsymbol{W}\boldsymbol{d^*} \Rightarrow (\boldsymbol{I} - \boldsymbol{W})\boldsymbol{d^*} = \mathcal{L}\boldsymbol{d^*} = 0. \tag{21}$$

Conditioned by the consistency with the observations, we have the following minimization problem:

$$\min_{\boldsymbol{d}} \|\widetilde{\boldsymbol{d}} - \boldsymbol{H}\boldsymbol{d}\|_2^2 + \lambda \|\mathcal{L}\boldsymbol{d}\|_2^2, \tag{22}$$

Note that model in Eq. (22) is exactly the same as the AR-based depth recovery model in Eq. (3) with the following identity: $\mathcal{L} = \boldsymbol{I} - \boldsymbol{Q}$, which establishes the connection between our AR-based depth recovery model and graph models.

The above discussions reveal that 1) our AR-based depth recovery model is essentially a global method enhanced by AR prediction error compensation, which inherits the advantages of this type of algorithms, e.g., achievable optimal solution and convenient mathematical analysis; 2) the derived algorithm to approximate the global optimum are a series of filtering on the observed depth maps and recovered ones with a feedback of AR prediction error, which enjoy the low-complexity of filtering based approaches; and 3) the proposed model is equivalent to a graph based model, which suggest that the AR-based depth recovery could be further improved via graph-model tools, e.g., graph transforms [52].

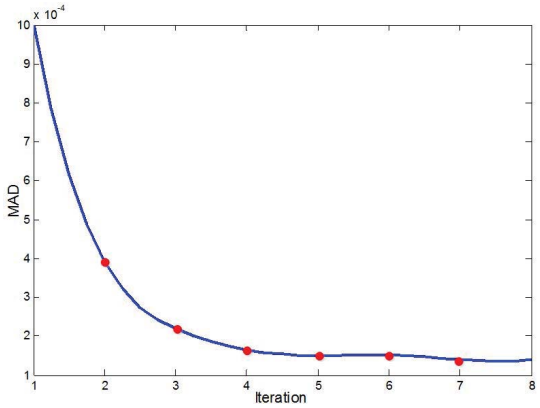


Fig. 3. The convergence curve of our fixed-point algorithm. Mean absolute difference (MAD) computed from the dataset captured in our lab is used for measuring the results of difference between two consecutive iterations.

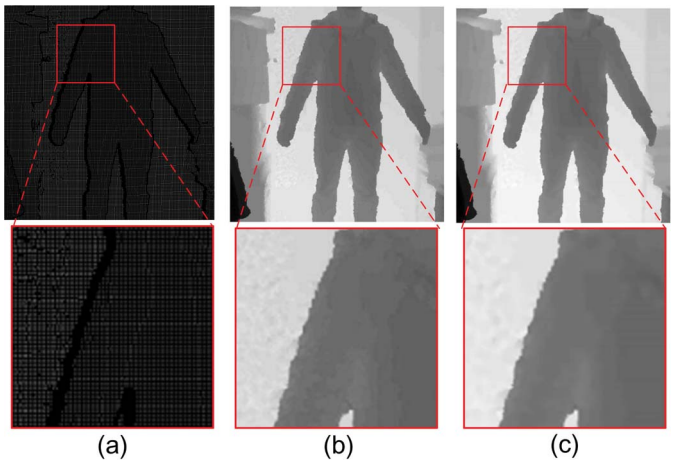


Fig. 4. Depth recovery results at: (a) the 1^{st} iteration, (b) the 2^{nd} iteration, and (c) the 4^{th} iteration. For easy observation, the image contrast is manually stretched to make the result more clearly, and the region in the red rectangular is enlarged.

VI. Experiments and Results

In this section, we first investigate the behavior of the proposed fixed-point iteration algorithm (Section VI-A), and then evaluate the depth recovery performance in two subsections: 1) Section VI-B: experiments on depth image recovery using Middlebury benchmark datasets with various synthetic degradations and some real captured depth images; 2) Section VI-C: experiments on multi-view depth video recovery using datasets captured by our stereo RGB-D acquisition system. The mean absolute difference (MAD) is used to measure the difference between two depth maps.

The parameters are set as follows: for both ARSDIR and ARMDVR algorithms, λ is set at 0.1 for high-fidelity depth inputs, and at larger values in the noisy cases (e.g., $\lambda = 10$ if the noise variance is around 15). The number of iterations is set to 4. In addition, for the temporal parameters in the ARMDVR algorithm, the size of the neighborhood $\mathcal{N}$ in Eq. (13) and the variance σ_3 in Eq. (15) are set at 9×9 and 3, respectively. The number of frames involved in spatiotemporal neighborhood is set to 2 for a reasonable performance-computation tradeoff.

TABLE I
THE OBJECTIVE COMPARISON (MAD) BETWEEN OUR PROPOSED ALGORITHM (ARSDIR) AND GLOBALAR [10]. THE RESULTS OF THREE KINDS OF TYPICAL DEGRADATIONS, I.E., UNDERSAMPLING (2×, 4×, 8×, 16×), TOF-LIKE DEGRADATION (8× UNDERSAMPLING WITH NOISE), AND KINECT-LIKE DEGRADATION (MISSING) ARE SHOWN IN THE TABLE

	ARSDIR						**GlobalAR**					
	Upsampling				ToF-like	Kinect-like	Upsampling				ToF-like	Kinect-like
	2×	4×	8×	16×	8×		2×	4×	8×	16×	8×	
Art	0.22	**0.46**	**0.63**	**1.52**	1.79	**0.49**	**0.18**	0.49	0.64	2.01	**1.70**	0.58
Book	**0.10**	**0.21**	0.40	**0.72**	**1.13**	**0.49**	0.12	0.22	**0.37**	0.77	1.15	0.53
Moebius	0.11	0.21	0.41	**0.74**	**1.10**	**0.53**	**0.10**	**0.20**	**0.40**	0.79	1.15	0.60
Reindeer	**0.16**	**0.34**	**0.57**	1.23	1.48	**0.64**	0.22	0.40	0.58	**1.00**	**1.28**	0.68
Laundry	**0.14**	**0.29**	**0.51**	**1.04**	1.49	**0.68**	0.20	0.34	0.53	1.12	**1.30**	0.75
Dolls	**0.13**	**0.25**	**0.50**	**1.05**	1.37	**0.67**	0.21	0.34	0.50	0.82	**1.32**	0.69
Average	**0.14**	**0.29**	**0.50**	**1.04**	1.39	**0.58**	0.17	0.33	**0.50**	1.09	**1.22**	0.64

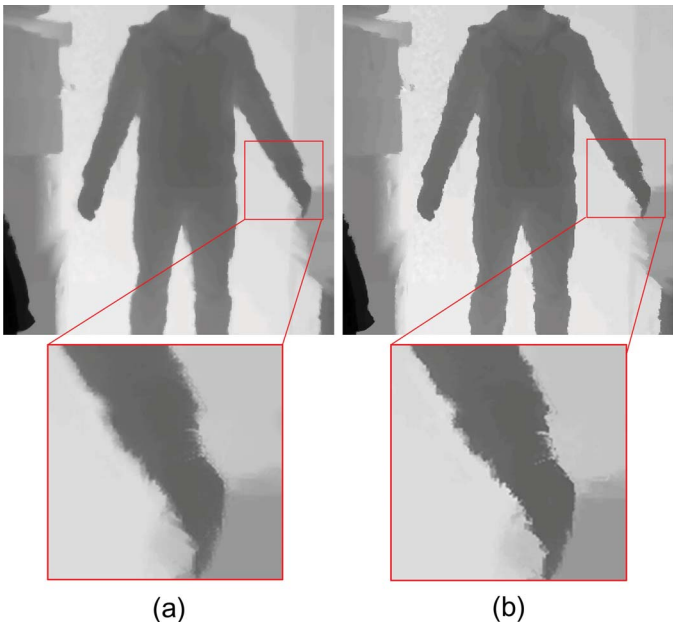


Fig. 5. Visual comparison between the recovered results using: (a) iterative filtering without residue feedback, (b) our ARSDIR (with residue feedback). The weighting schemes in both conditions are identical.

A. Evaluations of Convergence and Residue Feedback

1) Convergence Results: Fig. 3 shows MAD values between the recovered depth maps of two consecutive iterations, averaged over different depth maps captured by us. The MAD values decrease dramatically at the beginning, and turn to be stable after four iterations. Fig. 4 shows the visual results of recovered depth maps on the captured dataset *Standing* at the 1^{st}, 2^{nd}, and 4^{th} iterations, respectively. At the 2^{nd} iteration, all the missing depth pixels have been filled in, but the intense noise is still present in the recovered depth map; while at the 4^{th} iteration, most noise has been removed while preserving depth discontinuities. This demonstrates that our algorithm converges rapidly, generally after the fourth iteration for all the datasets. Experiments results in Section VI-B will further show that, with only four or five iterations, our iterative algorithm achieves comparative results to the global method [10].

2) Results on Residue Feedback: The significant difference between our proposed algorithms in Eq. (8) and other methods lies in the residue feedback derived through the fixed-point iteration algorithm of the recovery model. Fig. 5 shows the comparison between iterative filtering in Eq. (18) (without residue feedback) and our proposed algorithm (with residue feedback). Both methods use the same weighting scheme defined in Eq. (2). Visual results in Fig. 5 show that, thanks to the residue feedback, our proposed algorithm recovers sharper depth contours than the filtering scheme without residue feedback.

B. Experiments on Single-Image Depth Recovery

1) Results on Synthetic Datasets: Six datasets, *Art*, *Book*, *Moebius*, *Reindeer, Laundry,* and *Dolls* from the Middlebury's benchmark [53] are used for evaluation. The datasets with three kinds of typical degradations in [10],[2] i.e., undersampling, ToF-like degradation, Kinect-like degradation, are used for evaluation.

First, we compare the proposed iterative algorithm with the direct least squares solution (GlobalAR) in [10]. Both solve the same AR-based depth recovery model. Depth recovery results in MAD in Table I show that the proposed iterative algorithm generates better results than GlobalAR, especially for upsampling and the recovery of Kinect-like degradation.

More importantly, the running time of the proposed algorithm is much smaller than GlobalAR. As analyzed in [10], the superiority of (closed-loop) global methods is subject to higher computational complexity than the (open-loop) local filtering methods. The running time of most global methods is within the range of 10^0-10^1 minutes, while the local filtering methods require significantly less running time (about 10^0-10^1 seconds). Implemented in unoptimized MATLAB code and run on a desktop with a 3.4 GHz Core4 i7 CPU and 8 GB memory, the proposed algorithm takes about 6 seconds to recover a depth map to the size of 1920 × 1080 while the GlobalAR method takes about 4 minutes on average. Our proposed algorithm solves the same global objective function as GlobalAR,

[2]http://cs.tju.edu.cn/faculty/likun/projects/depth_recovery/index.htm

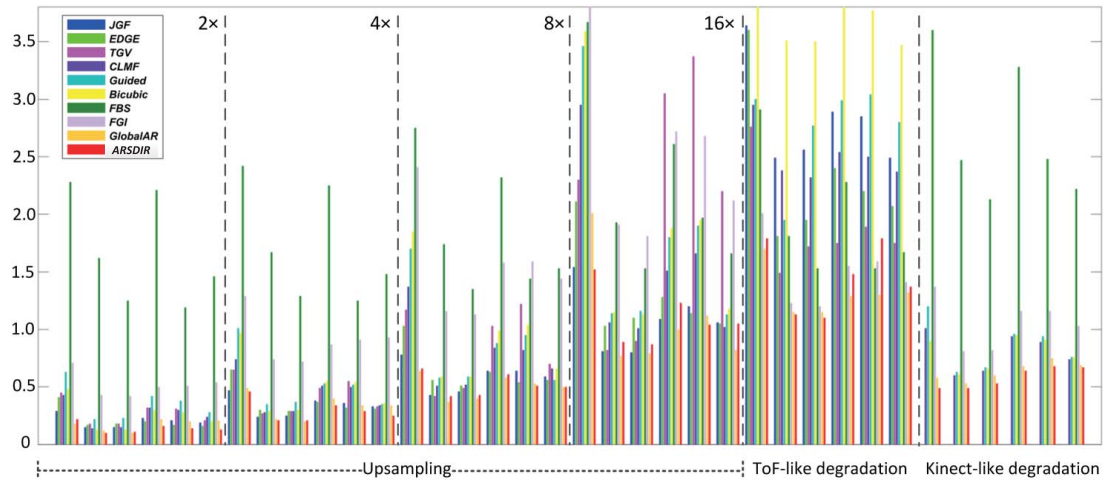


Fig. 6. Depth recovery results in MAD. From left to right, the recovery results of upsampling (2×, 4×, 8×, 16×), ToF-like degradation, and Kinect-like degradation are given successively, in which each group has compared using six datasets, i.e., *Art*, *Book*, *Moebius*, *Reindeer, Laundry,* and *Dolls*, from left to right. The compared methods are represented in different colors shown in the legend.

and also enjoys the low-complexity merit of local filtering methods.

Then, we compare our algorithm with other nine methods, including Bicubic interpolation, guided image filtering (Guided) [40], edge-weighted NLM-regularization (Edge) [9], cross-based local multipoint filtering (CLMF) [41], joint geodesic filtering (JGF) [11], total generalized variation (TGV) [27], fast bilateral solver (FBS) [42], fast global interpolation(FGI) [29], and GlobalAR [10]. Fig. 6 shows depth recovery results of eight methods on six datasets with various types of synthetic degradation. For compact presentation, results are visualized in a bar chart. From left to right, the recovery results of upsampling (2×, 4×, 8×, 16×), ToF-like degradation, and Kinect-like degradation are given successively, where the results of each group are for *Art*, *Book*, *Moebius*, *Reindeer, Laundry,* and *Dolls*, from left to right. In general, the results for higher upsampling factors (e.g., 16×) have relatively large MAD values due to more missing depth values in the measurements. Similarly, due to the mixture of noise and undersampling, the recovery results for ToF-like degradation also have large MAD values. On all the datasets, our proposed algorithm (ARSDIR, in red) has comparable recovery performance to GlobalAR as they solve the same model. It further outperforms other methods, which demonstrates the effectiveness of the proposed iterative algorithm with on-the-fly updating of AR coefficients.

2) Experiments on Real Datasets: Fig. 7 (a) shows two real datasets captured by Kinect v2 in our laboratory. The captured depth maps contain missing values caused by undersampling and structural holes along depth discontinuities, which are harder to recover than the regular missing values in pure upsampling. Structural artifacts around depth edges lead to ambiguity in locating the depth discontinuities and thus increase the difficulty of depth recovery.

We compare our ARSDIR model with FBS [42], FGI [29], and GlobalAR [10] on the two datasets. FBS and FGI are the latest depth recovery methods, and have great improvement in both speed and accuracy. FBS is a local filtering methods based on proposed fast bilateral solver, while FGI focuses on global modeling that decomposes the depth upsampling process into hierarchical global interpolation. Results are shown in Fig. 7 (b)∼(e). GlobalAR tends to generate some jaggy artifacts, while FGI brings slight over-smoothing results around edges. These two methods have results that are comparable to ours. However, FBS contains over-smoothing and texture-copying results due to rich color textures and the discontinuity mismatch between the color images and depth maps. With a carefully designed weighting scheme, our method achieves quite promising recovery quality particularly around depth discontinuities, and avoids the texture copying artifacts.

TABLE II
OBJECTIVE RESULTS (IN TERMS OF MAD IN MILLIMETERS) ON TOFMARK DATASETS

	Bicubic	Guided	CLMF	JGF	TGV	GlobalAR	ARSDIR
Books	16.23	15.74	13.89	17.39	12.36	12.25	**12.09**
Shark	17.78	18.21	15.10	18.17	15.29	14.71	**14.54**
Devil	16.66	27.04	14.55	19.02	14.68	13.83	**13.29**

Moreover, we also evaluate our algorithm on the *ToFMark* datasets [54] (*Books, Shark, Devil*). Table II presents the objective results in terms of recovery error measured by MAD in millimeter. Our method obtains the lowest recovery error for all the three test cases compared with other six methods. Visual results in Fig. 8 on *ToFMark Books* show that the proposed algorithm successfully recovers the structure of the stick and the cup while TGV [27] and GlobalAR [10] still present obvious artifacts.

C. Experiments on Multi-View Depth Images/Videos

The proposed ARMDVR model for multi-view depth video recovery extends the ARSDIR model by incorporating inter-frame and inter-view correlations. In this section, we test the performance on our ARMDVR model on both multi-view images and multi-view depth videos.

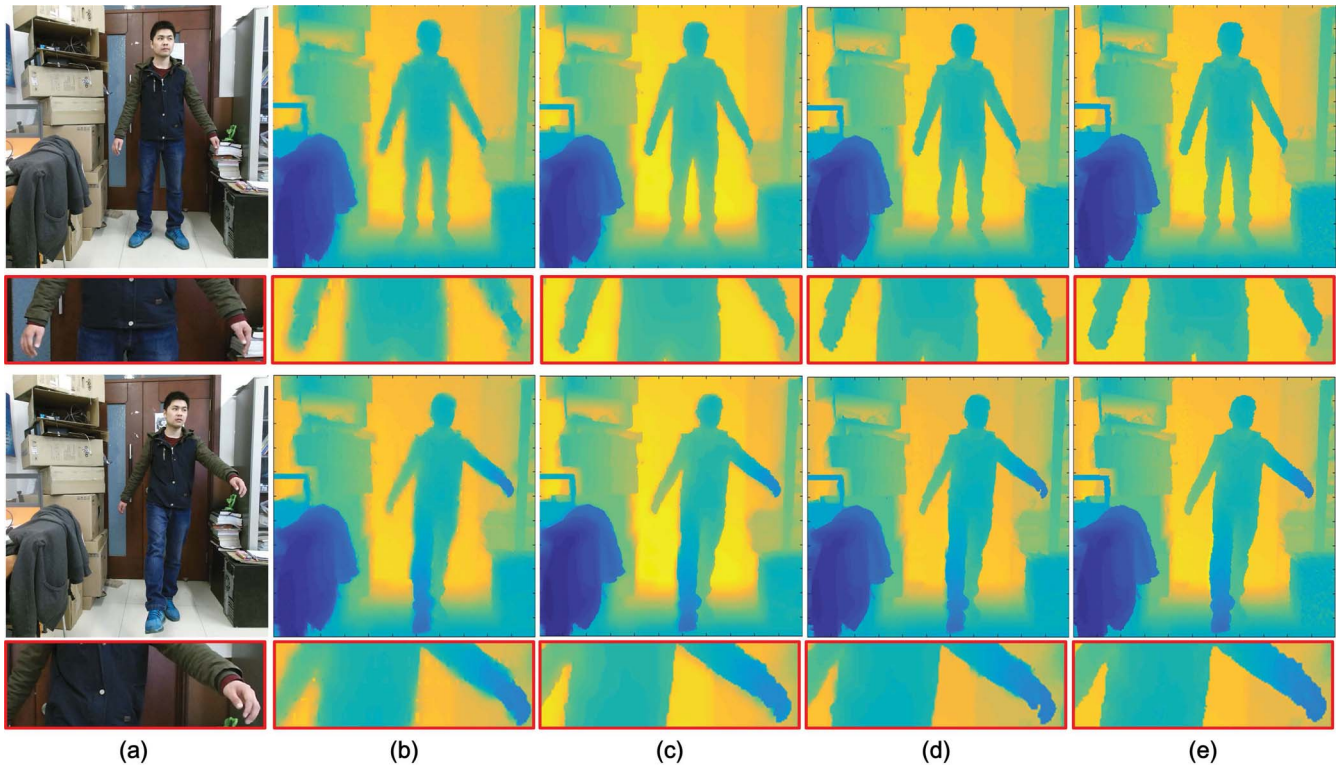


Fig. 7. Depth recovery results for two real datasets captured in our laboratory: (a) color image, recovered depth maps by (b) FBS [42], (c) FGI [29], (d) GlobalAR [10], and (e) ARSDIR. For visual inspection, regions highlighted by rectangles are enlarged.

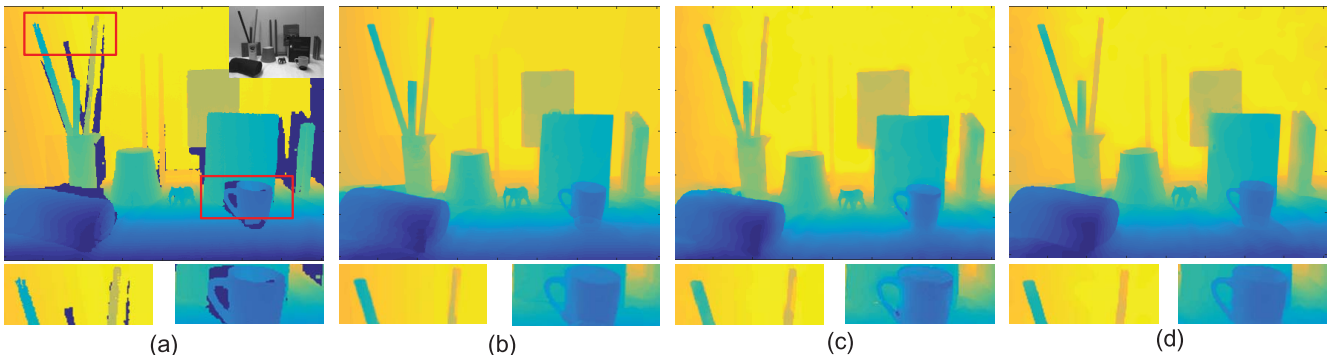


Fig. 8. Visual quality comparison on depth recovery for *Books* from *ToFMark* datasets: (a) Ground truth, (b) TGV [27], (c) GlobalAR [10], (d) Our method.

We set up a stereo RGB-D sensing system by using two Kinect v2 cameras, which could be considered as a prototype for 3DTV content generation using the multi-view video-plus-depth (MVD) representation. Note that our system is readily extendable by adding more Kinect v2 cameras for more robust sensing and wider angle of views. In our system, each RGB-D sensor has a frame rate of 30 frame/sec, and provides a low-resolution depth video stream and a high-quality color video stream. The resolutions of the color camera and depth camera are 1024×768 and 512×414, respectively. The stereo RGB-D camera rig is calibrated by the OpenCV calibration module [55], and the captured depth maps are warped to the viewpoint of the associated color camera.

Considering the length of Kinect v2 camera, the baseline distance between the two RGB-D cameras is set at 29 cm. Such a long baseline would cause more incorrect warping locations during calibration process, e.g., some background pixels would appear in the foreground areas after warping. To handle this, we remove some background pixels that likely belong to occluded areas by comparing the current depth value with the average of its neighboring pixels in a local window.

1) Results on Multi-View Images: Fig. 9 shows depth recovery results on three datasets captured by the stereo RGB-D camera rig. The stereo depth images are recovered by solving the AR-based multi-view depth recovery model in Eq. (9). For the recovery of each view, the warped depth image from the other view is also used as observed depth samples. Note that large areas with missing depth values appear around depth discontinuities because of occlusions. As a result, as shown in Fig. 9 (b) and (c), the fusion of multi-view depth images provides denser depth samples, which makes depth recovery less of an ill-posed problem. Figure 9 (d) and (e) show the recovery

Fig. 9. Multi-view depth maps recovery on real datasets: (a) color image; (b) the depth maps of current view; (c) the fused depth maps from multi-view depth observation; (d) recovered results using the depth maps in (b); (e) recovery results using fused depth maps in (c). For visual inspection, regions highlighted by rectangles are enlarged.

results from the observed depth maps in Fig. 9 (b) and (c), respectively. For all the three datasets, the results recovered from multi-view depth images have better quality than the recovery from single view alone. For example in the first dataset, without multi-view information, jaggy artifacts appear around the leg of the girl. In other two datasets, due to the large set of missing depth values (of width 20 missing pixels on average), the contours of the boys' head cannot be recovered correctly (in Fig. 9 (d)). On the contrary, as shown in Fig. 9 (e), the contour of the heads are correctly recovered thanks to the incorporation of interview information, which illustrates the effective of our extended AR-based depth recovery model with multi-view depth images.

2) Results on Multi-View Depth Videos: For evaluation, two stereo depth videos are captured by our RGB-D camera rig, named *Meeting* and *Yoga* as shown in Fig. 10. Consecutive 30 frames are used for each dataset. We compare our ARMDVR model in Eq. (11) with weighted mode filtering (WMF) [18], the state-of-the-art video super-resolution method. Besides, the AR-based multi-view depth image recovery model in Eq. (9) applied frame by frame is also compared to demonstrate the advantageous of the proposed ARMDVR model considering temporal information. The parameters in the WMF method are set according to [18].

The recovered results on the 3^{rd} and 6^{th} frames for *Yoga*, and the 8^{th} and 11^{th} frames for *Meeting*, are shown in Fig. 10. For the frame-wise application of AR-based multi-view depth image recovery, the results are subject to severe flickering artifacts in the background and particularly on the regions around depth discontinuities due to the ignorance of temporal consistency. For example, the contour of the man's head in *Meeting* is not consistent across the two frames, and the thin structure of handrail in *Meeting* can not be restored in 11^{th} frame. Both the proposed ARMDVR model and WMF integrate temporal information. However, WMF generates blurry and less temporally consistent results than ours, particularly around thin structures. Note that the rapid movement of the speaker's hand in *Meeting* leads to blurring in color frames. The simple optical flow used in WMF cannot estimate the motion accurately, which result in the ambiguity of depth estimation (the bottom row). Our method refines optical flow results to yield a more precise correspondence, and generates more temporally

Fig. 10. Multi-view depth videos recovery results on the dataset *Meeting* and *Yoga*. (a) and (b) shows 3^{rd} and 6^{th} frames picked from *Meeting*, while (c) and (d) shows 8^{th} and 11^{th} frames from *Yoga*. From top row to bottom row displays color frames, recovered results by spatial AR model (Eq. (9)), ARMDVR, and WMF [18], respectively.

consistent high-quality depth frames than WMF (shown in the third row).

Finally, we use the MAD values between neighboring frames as an objective metric to evaluate temporal consistency of the recovered results. Note that the difference originates from two parts, i.e., temporal flickering and object motion. Assuming that object motion is generally significant enough to be estimated, results with smaller MAD values are considered to have less flickering artifacts. Table III shows MAD values between 3^{th}-6^{th} frames for *Meeting* and 8^{th}-11^{th} frames for

TABLE III
OBJECTIVE COMPARISON (IN MAD) BETWEEN NEIGHBORING FRAMES ON 3^{RD}-6^{TH} FRAMES FROM *Meeting* AND 8^{TH}-11^{TH} FRAMES FROM *Yoga*

Index	Spatial AR		WMF		ARMDVR	
	Yoga	*Meeting*	*Yoga*	*Meeting*	*Yoga*	*Meeting*
1	0.525	1.096	0.225	0.510	**0.195**	**0.493**
2	0.532	0.923	0.278	0.381	**0.204**	**0.362**
3	0.528	0.844	0.230	0.442	**0.199**	**0.425**

Yoga, respectively. The results in Table III is consistent with the analysis above and with visual observation as well. The framewise AR-based multi-view depth image recovery model in Eq. (9) has the largest MAD values, corresponding to severe flickering artifacts, while our ARMDVR has the lowest MAD values for all cases, indicating temporally consistency.

VII. CONCLUSION

In this paper, we present the ARSDIR and ARMDVR frameworks for single depth image recovery and multi-view depth video recovery, respectively. We derive fixed-point iteration algorithms to efficiently solve the global AR-based depth recovery models. Besides, the connections between our model and other closely-related methods are discussed. Experimental results demonstrate that our algorithms inherit the advantage of global AR-based modeling, exploit low-complexity filtering-based approach with observation recurrence and residue feedback to approximate the global optimum, and could be analyzed and improved via graph-model tools.

Regarding the future work, an interesting direction is to explore more depth cues, such as object motion and stereo parallax, and to incorporate them into the proposed framework to improve the depth estimation result. Besides, we can also resort to some recent work on graph model, e.g., [52], to further improve the performance.

REFERENCES

[1] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, nos. 1–3, pp. 7–42, 2002.

[2] R. Szeliski *et al.*, "A comparative study of energy minimization methods for Markov random fields with smoothness-based priors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 6, pp. 1068–1080, Jun. 2008.

[3] C. Zhu and S. Li, "Depth image based view synthesis: New insights and perspectives on hole generation and filling," *IEEE Trans. Broadcast.*, vol. 62, no. 1, pp. 82–93, Mar. 2016.

[4] X. Ye, J. Yang, H. Huang, C. Hou, and Y. Wang, "Computational multi-view imaging with Kinect," *IEEE Trans. Broadcast.*, vol. 60, no. 3, pp. 540–554, Sep. 2014.

[5] Y. Kim, B. Ham, C. Oh, and K. Sohn, "Structure selective depth super-resolution for RGB-D cameras," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5227–5238, Nov. 2016.

[6] J. Lu and D. Forsyth, "Sparse depth super resolution," in *Proc. CVPR*, Boston, MA, USA, 2015, pp. 2245–2253.

[7] L. Wei, X. Chen, Y. Jie, and W. Qiang, "Robust color guided depth map restoration," *IEEE Trans. Image Process.*, vol. 26, no. 1, pp. 315–327, Jan. 2017.

[8] J. Diebel and S. Thrun, "An application of Markov random fields to range sensing," in *Proc. NIPS*, vol. 18. Vancouver, BC, Canada, 2005, pp. 291–298.

[9] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. Kweon, "High quality depth map upsampling for 3D-TOF cameras," in *Proc. ICCV*, 2011, pp. 1623–1630.

[10] J. Yang, X. Ye, K. Li, C. Hou, and Y. Wang, "Color-guided depth recovery from RGB-D data using an adaptive autoregressive model," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3443–3458, Aug. 2014.

[11] M.-Y. Liu, O. Tuzel, and Y. Taguchi, "Joint geodesic upsampling of depth images," in *Proc. CVPR*, 2013, pp. 169–176.

[12] M. Lindner, A. Kolb, and K. Hartmann, "Data-fusion of PMD-based distance-information and high-resolution RGB-images," in *Proc. ISSCS*, vol. 1, 2007, pp. 1–4.

[13] S. A. Guomundsson, R. Larsen, H. Aanæs, M. Pardas, and J. R. Casas, "TOF imaging in smart room environments towards improved people tracking," in *Proc. CVPR Workshops*, Anchorage, AK, USA, 2008, pp. 1–6.

[14] A. K. Riemens, O. P. Gangwal, B. Barenbrug, and R.-P. M Berretty, "Multi-step joint bilateral depth upsampling," in *Proc. VCIP*, 2009, pp. 1–12.

[15] A. Kolb, E. Barth, R. Koch, and R. Larsen, "Time-of-flight cameras in computer graphics," *Comput. Graph. Forum*, vol. 29, no. 1, pp. 141–159, 2010.

[16] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *Proc. CVPR*, 2015, pp. 370–378.

[17] M. Ni *et al.*, "Color-guided depth map super resolution using convolutional neural network," *IEEE Access*, vol. 5, pp. 26666–26672, 2017.

[18] D. Min, J. Lu, and M. N. Do, "Depth video enhancement based on weighted mode filtering," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1176–1190, Mar. 2012.

[19] L. Sheng, K. N. Ngan, and S. Li, "Temporal depth video enhancement based on intrinsic static structure," in *Proc. IEEE ICIP*, Paris, France, 2014, pp. 2893–2897.

[20] D. Kim, J. Choi, and K. Sohn, "Multiview ToF sensor fusion technique for high-quality depth map," in *Proc. SPIE*, vol. 8650. Burlingame, CA, USA, 2013, p. 865006.

[21] Q. Liu, Z. Zha, and Y. Yang, "Gradient-domain-based enhancement of multi-view depth video," *Inf. Sci.*, vol. 281, pp. 750–761, Oct. 2014.

[22] J. Choi, D. Min, and K. Sohn, "Reliability-based multiview depth enhancement considering interview coherence," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 4, pp. 603–616, Apr. 2014.

[23] W. Hannemann, A. Linarth, B. Liu, and G. Kokai, "Increasing depth lateral resolution based on sensor fusion," *Int. J. Intell. Syst. Technol. Appl.*, vol. 5, nos. 3–4, pp. 393–401, 2008.

[24] J. Lu, D. Min, R. S. Pahwa, and M. N. Do, "A revisit to MRF-based depth map super-resolution and enhancement," in *Proc. ICASSP*, 2011, pp. 985–988.

[25] B. Huhle, S. Fleck, and A. Schilling, "Integrating 3D time-of-flight camera data and high resolution images for 3DTV applications," in *Proc. 3DTV Conf.*, 2007, pp. 1–4.

[26] J. Zhu, L. Wang, J. Gao, and R. Yang, "Spatial-temporal fusion for high accuracy depth maps using dynamic MRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 899–909, May 2010.

[27] D. Ferstl, C. Reinbacher, R. Ranftl, M. Rüther, and H. Bischof, "Image guided depth upsampling using anisotropic total generalized variation," in *Proc. ICCV*, Sydney, NSW, Australia, 2013, pp. 993–1000.

[28] J. Yang, X. Ye, K. Li, and C. Hou, "Depth recovery using an adaptive color-guided auto-regressive model," in *Proc. ECCV*, 2012, pp. 158–171.

[29] Y. Li, D. Min, M. N. Do, and J. Lu, "Fast guided global interpolation for depth and motion," in *Proc. ECCV*, 2016, pp. 717–733.

[30] Y. Zuo, Q. Wu, J. Zhang, and P. An, "Explicit edge inconsistency evaluation model for color-guided depth map enhancement," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 2, pp. 439–453, Feb. 2018.

[31] L. Pizarro, P. Mrázek, S. Didas, S. Grewenig, and J. Weickert, "Generalised nonlocal image smoothing," *Int. J. Comput. Vis.*, vol. 90, no. 1, pp. 62–87, 2010.

[32] Q. Yang, R. Yang, J. Davis, and D. Nistér, "Spatial-depth super resolution for range images," in *Proc. CVPR*, Minneapolis, MN, USA, 2007, pp. 1–8.

[33] J. Dolson, J. Baek, C. Plagemann, and S. Thrun, "Upsampling range data in dynamic environments," in *Proc. CVPR*, San Francisco, CA, USA, 2010, pp. 1141–1148.

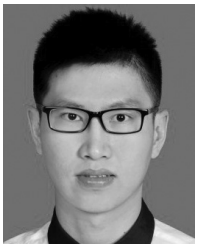
[34] B. Huhle, T. Schairer, P. Jenke, and W. Straßer, "Fusion of range and color images for denoising and resolution enhancement with a non-local filter," *Comput. Vis. Image Understanding*, vol. 114, no. 12, pp. 1336–1345, 2010.

[35] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, "Joint bilateral upsampling," *ACM Trans. Graph.*, vol. 26, no. 3, p. 96, 2007.

[36] F. Li, J. Yu, and J. Chai, "A hybrid camera for motion deblurring and depth map super-resolution," in *Proc. CVPR*, Anchorage, AK, USA, 2008, pp. 1–8.
[37] Q. Yang, K.-H. Tan, B. Culbertson, and J. Apostolopoulos, "Fusion of active and passive sensors for fast 3D capture," in *Proc. MMSP*, St-Malo, France, 2010, pp. 69–74.
[38] D. Chan, H. Buisman, C. Theobalt, and T. Sebastian, "A noise-aware filter for real-time depth upsampling," in *Proc. ECCV Workshop M2SFA2*, 2008, pp. 1–12.
[39] J. Lei *et al.*, "Depth map super-resolution considering view synthesis quality," *IEEE Trans. Image Process.*, vol. 26, no. 4, pp. 1732–1745, Apr. 2017.
[40] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Proc. ECCV*, 2010, pp. 1–14.
[41] J. Lu, K. Shi, D. Min, L. Lin, and M. N. Do, "Cross-based local multipoint filtering," in *Proc. CVPR*, Providence, RI, USA, 2012, pp. 430–437.
[42] J. T. Barron and B. Poole, "The fast bilateral solver," in *Proc. ECCV*, 2016, pp. 617–632.
[43] F. Duan, "Spatio-temporal consistency in stereoscopic video depth map sequence estimation," *J. Inf. Comput. Sci.*, vol. 11, no. 18, pp. 6497–6508, 2014.
[44] Q. Wang, S. Li, H. Qin, and A. Hao, "Super-resolution of multi-observed RGB-D images based on nonlocal regression and total variation," *IEEE Trans. Image Process.*, vol. 25, no. 3, pp. 1425–1440, Mar. 2016.
[45] J. Zhang *et al.*, "A unified scheme for super-resolution and depth estimation from asymmetric stereoscopic video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 3, pp. 479–493, Mar. 2016.
[46] B. Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed. Philadelphia, PA, USA: SIAM, 2003.
[47] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. IJCAI*, vol. 81. Vancouver, BC, Canada, 1981, pp. 674–679.
[48] C. Liu and W. T. Freeman, "A high-quality video denoising algorithm based on reliable motion estimation," in *Proc. ECCV*, 2010, pp. 706–719.
[49] A. Buja, T. Hastie, and R. Tibshirani, "Linear smoothers and additive models," *Ann. Stat.*, vol. 17, no. 2, pp. 453–510, 1989.
[50] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. ICCV*, 1998, pp. 839–846.
[51] T. Brox, O. Kleinschmidt, and D. Cremers, "Efficient nonlocal means for denoising of textural patterns," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1083–1092, Jul. 2008.
[52] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.
[53] *Middlebury Datasets*. Accessed: Jun. 1, 2017. [Online]. Available: http://vision.middlebury.edu/stereo/data/
[54] *ToFmark Datasets*. Accessed: Jun. 26, 2017. [Online]. Available: http://rvlab.icg.tugraz.at/tofmark/
[55] *Open Source Computer Vision (OpenCV)*. Accessed: May 15, 2017. [Online]. Available: http://opencv.org

Jingyu Yang (M'10–SM'17) received the B.E. degree from the Beijing University of Posts and Telecommunications in 2003, and the Ph.D. degree (Hons.) from Tsinghua University in 2009.

He has been a Faculty Member with Tianjin University, China, since 2009, where he is currently a Research Professor with the School of Electrical and Information Engineering. He was with Microsoft Research Asia (MSRA) in 2011, within the MSRA's Young Scholar Supporting Program, and the Signal Processing Laboratory, EPFL, Lausanne, Switzerland, in 2012, and from 2014 to 2015. His research interests include image/video processing, 3-D imaging, and computer vision. He has authored or co-authored over 70 high quality research papers (including dozens of IEEE TRANSACTIONS and top conference papers). He was a recipient of the Best 10% Paper Award in IEEE VCIP 2016 for his co-authored paper and the Platinum Best Paper Award in IEEE ICME 2017. He served as the Special Session Chair in VCIP 2016 and the Area Chair in ICIP 2017. He was selected into the program for New Century Excellent Talents in University from the Ministry of Education, China, in 2011, the Reserved Peiyang Scholar Program of Tianjin University in 2014, and the Tianjin Municipal Innovation Talent Promotion Program in 2015.



Xinchen Ye (M'18) received the B.E. and Ph.D. degrees from Tianjin University, Tianjin, China, in 2012 and 2016, respectively. He was with the Signal Processing Laboratory, EPFL, Lausanne, Switzerland, in 2015 under the Grant of the Swiss Federal Government.

He has been a Faculty Member with the Dalian University of Technology, Dalian, China, since 2016, where he is currently a Assistant Professor with the DUT-RU International School of Information Science and Engineering. His current research interests include image/video processing, 3-D imaging, medical image processing, and computer vision. He was a recipient of the Platinum Best Paper Award in the IEEE ICME 2017 for his co-authored paper.

Pascal Frossard (S'96–M'01–SM'04–F'18) received the M.S. and Ph.D. degrees in electrical engineering from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 1997 and 2000, respectively. From 2001 to 2003, he was a member of the Research Staff with the IBM T. J. Watson Research Center, Yorktown Heights, NY, USA, where he researched on media coding and streaming technologies. Since 2003, he has been a Faculty Member with EPFL, where he heads the Signal Processing Laboratory. His research interests include graph signal processing, image representation and coding, visual information analysis, and distributed signal processing and communications.

He was a recipient of the Swiss NSF Professorship Award in 2003, the IBM Faculty Award in 2005, the IBM Exploratory Stream Analytics Innovation Award in 2008, the Google Faculty Award 2017, the IEEE TRANSACTIONS ON MULTIMEDIA Best Paper Award in 2011, and the *IEEE Signal Processing Magazine* Best Paper Award 2016. He has been the General Chair of IEEE ICME 2002 and Packet Video 2007. He has been the Technical Program Chair of IEEE ICIP 2014 and EUSIPCO 2008, and a member of the organizing or technical program committees of numerous conferences. He has been a Senior Area Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING since 2015, an Associate Editor of the IEEE TRANSACTIONS ON BIG DATA since 2015, the IEEE TRANSACTIONS ON IMAGE PROCESSING from 2010 to 2013, the IEEE TRANSACTIONS ON MULTIMEDIA from 2004 to 2012, and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2006 to 2011. He was an Elected Member of the IEEE Multimedia Signal Processing Technical Committee from 2004 to 2007 and since 2016, the IEEE Visual Signal Processing and Communications Technical Committee since 2006, and the IEEE Multimedia Systems and Applications Technical Committee since 2005. He has served as the Chair of the IEEE Image, Video and Multidimensional Signal Processing Technical Committee from 2014 to 2015, and the Steering Committee Chair from 2012 to 2014 and the Vice-Chair from 2004 to 2006 of the IEEE Multimedia Communications Technical Committee.