In this article, we discuss enhanced full 360° 3D reconstruction of dynamic scenes containing non-rigidly deforming objects using data acquired from commodity depth or 3D cameras. Several approaches for enhanced and full 3D reconstruction of non-rigid objects have been proposed in the literature. These approaches suffer from several limitations due to requirement of a template, inability to tackle large local deformations and topology changes, inability to tackle highly noisy and low-resolution data, and inability to produce online results. We target online and template-free enhancement of the quality of noisy and low-resolution full 3D reconstructions of dynamic non-rigid objects. For this purpose, we propose a view-independent recursive and dynamic multi-frame 3D super-resolution scheme for noise removal and resolution enhancement of 3D measurements. The proposed scheme tracks the position and motion of each 3D point at every timestep by making use of the current acquisition and the result of the previous iteration. The effects of system blur due to per-point tracking are subsequently tackled by introducing a novel and efficient multi-level 3D bilateral total variation regularization. These characteristics enable the proposed scheme to handle large deformations and topology changes accurately. A thorough evaluation of the proposed scheme on both real and simulated data is carried out. The results show that the proposed scheme improves upon the performance of the state-of-the-art methods and is able to accurately enhance the quality of low-resolution and highly noisy 3D reconstructions while being robust to large local deformations.

CCS Concepts: • Computing methodologies → Reconstruction; 3D imaging; Active vision;

Additional Key Words and Phrases: 3D reconstruction, super-resolution, non-rigid registration, point-cloud enhancement, 3D bilateral total variation, 3D point tracking

ACM Reference format:
Hassan Afzal, Djamila Aouada, Bruno Mirbach, and Björn Ottersten. 2018. Full 3D Reconstruction of Non-Rigidly Deforming Objects. ACM Trans. Multimedia Comput. Commun. Appl. 14, 1s, Article 24 (March 2018), 23 pages.
https://doi.org/10.1145/3177756
1 INTRODUCTION

Acquiring high quality and complete 360° 3D reconstructions of dynamic scenes containing non-rigidly deforming objects is one of the fundamental goals of research in computer vision and robotics. Such reconstructions can be effective in solving various problems in the domains of security and surveillance, virtual reality, gaming, and so on.

Reconstruction of a 3D world has traditionally been achieved via photometric cameras by correlating the same 3D points in 2D images across different views. Non-rigid objects have been reconstructed using both mono-view systems [27, 66], which provide partial coverage of the scene, and multi-view systems, which provide complete coverage of the scene instantaneously [15, 20, 58]. Due to the limitations of photometric-sensing systems, most of these methods usually require expensive and highly constrained setups [59]. They either use pre-built templates of target objects [14, 56, 58], or build them as a first step [20, 26, 58, 66] as shape priors for, e.g., non-rigid tracking, hole filling, shape completion, and so forth.

Commodity depth cameras such as Microsoft Kinect v1 and v2 [41], Asus Xtion Pro Live [9], and PMD camboard nano [49], have opened further the possibilities of research in this domain by providing 2.5D information that can directly be converted into 3D point clouds. This, on one hand, diminishes barriers on highly constrained setups but, on the other hand, poses challenges due to noisy and, in some cases, low-resolution (LR) acquired measurements [49]. Hence, the goal of acquiring high quality and complete 3D reconstructions of non-rigidly deforming objects still remains unfulfilled.

Recently, researchers have tried to overcome these challenges by focusing on building complete and enhanced 3D reconstructions of non-rigidly deforming objects using commodity depth cameras. Apart from template-based approaches [22, 65, 68, 70], there are mono-view methods that build complete and enhanced 3D reconstructions by incrementally fusing temporal information [18, 64, 67]. Such methods are view-dependent, and therefore cannot directly be used to provide full 3D reconstructions instantaneously [3, 45]. View-independent temporal fusion-based reconstruction methods such as VI-KinectDeform have also been proposed but they do not target LR measurements [1]. RecUP-SR, on the other hand, targets quality and resolution enhancement of 3D reconstructions using depth maps only, but is restricted to mono-view partial reconstructions and may not be robust to fast and abrupt 3D motions [33, 34].

Recently proposed multi-view, systems-based, template-free methods that provide complete and enhanced reconstructions of non-rigidly deforming objects instantaneously are also not robust to LR depth measurements. Moreover, they are either restricted to limited deformations [38, 57, 64, 67] or are sensitive to certain topology changes [21, 52].

In this article, we propose to tackle several challenges that lie in the way of achieving online, high quality, and complete 3D reconstructions of scenes, containing non-rigidly deforming objects with little constraints on their topology and motion. We base our work on measurements acquired via commodity depth cameras due to their low cost, flexibility, and instantaneous capture of 3D information, but the challenges that arise due to LR and noisy measurements of these cameras need to be tackled as well. For this purpose, we propose a view-independent recursive and dynamic multi-frame 3D super-resolution (SR) scheme. This scheme targets enhancement of resolution and quality of noisy LR 3D measurements, of non-rigidly deforming objects, acquired by commodity depth sensors.

The proposed approach is template-free and works directly on 3D points. This gives it flexibility to the types of objects being reconstructed, and the ability to capture their characteristics, i.e., position and motion in the 3D world more accurately. The affects of system blur are tackled via a novel and efficient multi-level 3D bilateral total variation (BTV) regularization. To our knowledge,
the proposed algorithm is the first view-independent, recursive, and dynamic multi-frame 3D SR method that targets complete 3D reconstruction of scenes/objects. The recursive nature of this algorithm allows it to produce enhanced high-resolution (HR) point clouds at each timestep by taking as input only the current noisy and LR measurement and the resulting noise-free HR point cloud obtained at the previous timestep. The pipeline of the proposed algorithm is shown in Figure 1, and details follow.

1.1 Contributions

We propose a novel recursive and dynamic multi-frame 3D SR scheme for producing 3D videos containing enhanced and complete 3D reconstructions of non-rigidly deforming objects:

- The proposed scheme produces HR, enhanced, and complete 3D reconstructions recursively by fusing the current acquisition, from a depth/3D camera system, and the result of previous iteration.
- It is a view-independent, and template-free, resolution and quality-enhancement scheme based on per-point tracking in 3D space, which allows it to be robust to changes in topology, and large motions.
- We have formulated this scheme to handle per-coordinate independent as-well-as depth camera specific noise in the acquired 3D points.
- A novel and efficient multi-level 3D BTV regularization is also proposed. It is used to handle system blur and correct per-point position and motion estimates, at every iteration.
- Detailed experimental, quantitative, and qualitative evaluations have been carried out using both simulated and real data. Results show that the proposed dynamic scheme outperforms the state-of-the-art filtering algorithms and produces accurate, smooth, and feature-preserving 3D reconstructions.

1.2 Article Overview

This article is organized as follows: We start by giving a brief over-view of state-of-the-art methods for acquiring complete and enhanced 3D reconstructions of non-rigidly deforming objects using depth cameras in Section 2. Section 3 formulates the problem of recursive dynamic multi-frame 3D SR. It is followed by details of the proposed algorithm, which solves the given problem by tracking and filtering the position and motion of each 3D point in LR upsampled measurements affected by per-coordinate independent as well as depth camera dependent noise in Section 4. After per-point
tracking, a smoothing and deblurring step is required, at each timestep, for 3D point position and motion estimates. For this purpose, a novel 3D BTV regularization is proposed. In Section 5, results of the proposed approach based on qualitative and quantitative experiments in comparison with state-of-the-art methods are presented, which is followed by a conclusion in Section 6.

2 RELATED WORK

In this section, we review the state of the art related to enhancement of 3D dynamic videos containing non-rigidly deforming objects. Our focus will be on depth-camera-based techniques that are used to acquire complete and noise-free 3D reconstructions of non-rigidly deforming objects.

Compared to photometric cameras, commodity 3D-camera-based reconstruction approaches, although aided by 3D acquisitions, have to overcome problems related to noise and limited resolution. After the advent of commodity RGB-D or 3D-camera-based enhanced 3D reconstruction techniques for rigid objects [11, 19, 46, 50], researchers have moved toward handling non-rigid deformations by proposing to construct complete and enhanced 3D models of mainly human subjects by fusing information from multiple views. This requires handling quasi-rigid motions between different views for which a global non-rigid registration is performed [38, 57], or a model-to-part registration based on deformation graph [53] or Shape Completion and Animation of People (SCAPE) model [5] is used to avoid error accumulation [63, 67]. The works of Cui et al. [17] and Shapiro et al. [24] are interesting in this regard, as they try to tackle the limited-resolution of the data acquired from commodity 3D cameras as well. Before data fusion, a resolution enhancement step, SR, is performed on data from individual views with the help of either HR RGB images [17] or mono-view filtering under rigidity constraints [24, 46] to get enhanced HR 3D reconstructions.

To efficiently achieve enhanced 3D reconstructions of non-rigid objects, undergoing relatively large local motions, template-based methods have been proposed in which a high-quality template is built as a first step. Li et al. [38] and Zollhöfer et al. [70] propose to pre-build high-quality, complete templates of the target objects, which are then used to track non-rigid deformations before being fused with current measurements to produce enhanced 3D reconstructions. These methods are restricted to the class of objects that can stay static or undergo controlled rigid motions for a sufficient period of time for accurate template reconstruction.

On the other hand, methods based on different 3D non-rigid registration algorithms, using compact deformable parameterizations based on, e.g., deformation graphs [37, 53], thin plate splines [10, 13], skeleton extraction [58], consensus, and matching under articulated motion assumptions [69] have been proposed [16, 40]. Ye et al. propose a performance capture method for complete human bodies based on skeleton fitting with three hand-held Kinect v1 cameras by making use of RGB information to aid in the registration process [65]. Li et al. [38] employ a visual hull prior, with pair-wise non-rigid scan registration based on deformation graphs [37] for hole-filling and shape completion based on relatively noise-free data.

Another class of template-free methods for complete reconstruction of 3D objects is based on spatio-temporal refinement and tracking of input data to build 4D models offline [43, 54]. Wand et al. use a topology-aware adaptive sub-space deformation technique to reduce the drift, together with as-rigid-as-possible and temporally coherent constraints on motion, to establish correspondences between acquisitions in 3D videos [60, 61]. The computed deformation field is used to construct a noise-free template from partial acquisitions. Sharf et al. relax the motion and spatial coherence constraints by using a bounded volume [52]. Their method suffers from flickering effects while still not being able to capture large deformations [38]. A recent work by Xu et al. [64] is interesting wherein a complete 3D model, and ultimately a 4D reconstruction, is iteratively built by fusing the non-rigidly deforming partial and LR observations and parameters of deformation subspace with the help of the Coherent Point Drift (CPD) algorithm [44]. CPD is a probabilistic
non-rigid registration algorithm which is shown to handle arbitrary motions and arbitrary topologies accurately. The method of Xu et al. also has a tendency to suffer from drift due to large deformations.

Similar to Xu et al. [64], a recent body of work in this domain uses a recursive approach for temporal fusion and incremental construction of high-quality 3D reference models without the need to build complete 4D reconstructions. In this vain, Dou and Fuchs have proposed a recursive template-free scheme, using a multi-view system composed of 10 Kinect v1 cameras, which track the motion of dynamic human subjects using deformation graphs [21]. After motion estimation, partial measurements and the reference frame are fused together using a directional distance function to produce enhanced 3D reconstructions [21, 22]. This method is restricted by the limitations of having open gesture topology for the reference frame. Moreover, the results lack quantitative analysis, and the technique has not been tested in setups with fewer cameras or with LR acquisitions. DynamicFusion is a similar work which targets real-time enhancement and incremental surface completion of non-rigidly deforming objects using a mono-view system, but suffers from similar limitations as the work by Dou and Fuchs [21, 45]. VolumeDeform builds upon DynamicFusion by improving the non-rigid registration via computation of local deformations at a finer scale together with using sparse RGB features to reduce drift and improve loop closures [29].

To tackle the above-mentioned challenges of recursive surface enhancement techniques, there are recent methods proposed by Afzal et al., namely KinectDeform [3] and VI-KinectDeform [1]. They are able to handle large local motions and do not require a reference model with a fixed topology. KinectDeform is a view-dependent method and hence can only produce partial reconstructions. VI-KinectDeform, on the other hand, is a view-independent moving least squares (MLS) and Kalman filter based, 3D video enhancement scheme which could directly be used in 3D multi-view systems. It has duly been tested for mono-view systems but has not been tested for and may not perform well on LR data [1].

To tackle LR and noisy non-rigidly deforming data we look into image-based SR techniques [6–8, 31–34]. It is important to mention the work of Al Ismaeil et al. in this regard, which, though restricted to enhancement of mono-view dynamic depth videos, proposes to tackle the problem of LR sensing systems via a recursive dynamic multi-frame depth SR algorithm [33, 34]. This algorithm recursively estimates an HR and enhanced depth map at each timestep, by taking as input the current upsampled LR measurement and the result of previous timestep to track and correct the depth and radial displacement values of each 3D point, associated with a pixel, using a Kalman filter [35]. This method performs well on various non-rigid scenes but cannot be used for full 3D reconstructions. Moreover, due to range flow approximation, this method can face difficulties to track fast and abrupt motions. Mac Aodha et al. have proposed a learning-based SR approach known as patch-based single image SR (SISR) [6]. This approach targets resolution enhancement of LR image patches using a dictionary of noise-free synthetic HR patches. Bondi et al. [12], on the other hand, have proposed a 3D SR method that targets HR and noise-free 3D reconstruction of human faces for improved recognition. The 3D data, from LR mono-view dynamic depth videos, is accumulated via non-rigid registration based on CPD algorithm. The accumulated data is then used to estimate the 2D manifold of the face to get HR enhanced 3D reconstructions.

This overview of the state of the art suggests that, although several approaches for enhanced and complete 3D reconstructions of non-rigid objects undergoing local motions have been proposed, they suffer from several limitations. These limitations are due to the requirements for template generation, inability to tackle large deformations, inability to tackle highly noisy and LR data, and inability to produce online results.

To tackle these limitations, we propose a template-free and recursive SR approach capable of handling highly noisy and LR 3D data acquired via commodity depth/3D cameras. The pipeline
of the proposed algorithm is shown in Figure 1. Following image-based SR approaches [32, 33], at every timestep, it upsamples the acquired measurement and uses it together with the result of previous timestep to track and correct the position and motion of each 3D point. It, therefore, avoids error accumulation or drift caused by large deformations. Furthermore, regularization of positions and correction of motion is carried out, at each timestep, with the help of a novel 3D BTV regularization. Working directly with 3D points allows the proposed scheme to be view-independent. This enables the proposed scheme to produce high-quality full 3D reconstructions of dynamic scenes by making it generic to the number of cameras used in 3D acquisition systems. We validate the proposed approach via quantitative and qualitative analysis on simulated and real data.

3 PROBLEM FORMULATION

A 3D acquisition system captures a full 360° LR 3D video \( \{ L_t \} \) of a scene containing non-rigidly deforming objects, with each unorganized point cloud represented as an ordered point-set \( L_t \), acquired at time \( t \), and containing \( M \) 3D points, where \( M \in \mathbb{N}^* \). The acquired points in \( L_t \) approximate the underlying surface of objects in the scene. The objective is to reconstruct an enhanced HR 3D video \( \{ H_t \} \) where each point-set \( H_t = \{ p_1^t, \ldots, p_U^t \} \). Each point \( p_i^t = (x_i^t, y_i^t, z_i^t)\top \) where \( x_i^t, y_i^t \) and \( z_i^t \in \mathbb{R} \), \( \top \) is the transpose, and \( i \in \{1, \ldots, U\} \). Also, \( U = o \times M \), where \( o \in \mathbb{N}^* \) is the factor by which the resolution of the input data is enhanced. It is also known as the SR factor.

Let us assume that each LR acquired point cloud \( L_t \) is related to the corresponding HR cloud \( H_t \) via the sensor model

\[
L_t = r(H_t) + W_t,
\]

where \( r(\cdot) \) is the measurement function that incorporates system blur and downsampling operators, and \( W_t \) represents additive white noise at time \( t \) and has same size as \( L_t \). We can perform dense upsampling on the acquired LR point clouds as a pre-processing step, which eliminates the resolution difference between the measured data and the desired \( \hat{H}_t \) that we are to estimate, and helps in decreasing the registration error [31, 32]. Considering a dense upsampling operator \( \uparrow \) that performs an increase or enhancement in resolution, with a factor \( o \), Equation (1) becomes

\[
\hat{H}_t = L_t \uparrow = [r(H_t)] \uparrow + W_t \uparrow.
\]

Moreover, each HR point cloud \( H_{t-1} \) undergoes a dynamic deformation at time \( t \) to give the HR point cloud \( H_t \) via

\[
H_t = h_t(H_{t-1}) + \mathcal{F}_t,
\]

where \( h_t(\cdot) \) is the local deformation function which deforms \( H_{t-1} \) to \( H_t \), and \( \mathcal{F}_t \) is the innovation containing information about new and disappearing points [33, 34].

The objective of this article is to devise an algorithm that recursively estimates \( H_t \) by taking into account the current upsampled input point cloud \( \hat{H}_t \), the previous result \( \hat{H}_{t-1} \), and the estimated 3D non-rigid deformation relating them, such that

\[
\hat{H}_t = \begin{cases} 
\hat{H}_t & \text{for } t = 0, \\
\text{filt}(\hat{H}_{t-1}, \hat{H}_t) & \text{for } t > 0,
\end{cases}
\]

where \( \text{filt}(\cdot, \cdot) \) is a filtering function that mitigates the effects of cameras’ measurement limitations, which result in noisy measurements with limited resolution and system blur.
4 PROPOSED APPROACH

4.1 Overview

In this section, we present a solution to the problem formulated in Section 3 by proposing a view-independent recursive dynamic multi-frame 3D SR algorithm. Figure 1 gives an overview of this algorithm. After upsampling the acquired LR point cloud \( L_t \) to get \( \tilde{H}_t \), using Equation (2), we estimate the non-rigid deformations which register the enhanced HR result of previous iteration \( \tilde{H}_{t-1} \) with \( \tilde{H}_t \). This registration is used to establish point-to-point correspondences between \( \tilde{H}_t \) and \( \tilde{H}_{t-1} \), which allows us to track and filter the position and motion of each point in \( \tilde{H}_t \). For this purpose, we use the CPD algorithm [44], which is a probabilistic method, wherein the matching of two point clouds is considered a probability density estimation problem [44]. The CPD algorithm non-rigidly registers \( \tilde{H}_{t-1} \) to \( \tilde{H}_t \), which is followed by a nearest-neighbor search for establishing point-to-point correspondences. For per-point refinement via tracking, in this article, we use a Kalman filter [35], which performs prediction and correction for each 3D point’s motion and position using the point-to-point correspondence information. This results in a noise-free but blurred estimate of \( \tilde{H}_t \) [51]. We use a novel 3D BTV regularization to perform deblurring and produce a noise-free HR estimate \( \hat{H}_t \). After that, a motion correction step using updated point positions in \( \hat{H}_t \) is also carried out. These steps are repeated for every measurement \( L_t \). This results in a recursive filtering process, as formulated in Equation (4), which enhances the resolution and quality of \( L_t \) using the previous result.

In what follows, we describe the method for per-point tracking using the correspondence information provided by the non-rigid registration algorithm. After that we describe the proposed 3D BTV regularization based deblurring and correction method.

4.2 Per-Point Refinement via Tracking

For simplification of notation, in what follows we remove the point indices \( i \), i.e., \( r'_t = r_t, \forall r'_t \in \mathbb{R}^3 \). We assume that the non-rigid registration step, in Figure 1, establishes point-to-point correspondences between the points \( \tilde{p}_t \) and \( \tilde{p}_{t-1} \). Now the measurement model for each point follows from Equation (2) such that

\[
\tilde{p}_t = p_t + n_t,
\]

where \( n_t = (n(x,t), n(y,t), n(z,t))^T \) represents per-coordinate independent Gaussian noise that affects each measured point \( \tilde{p}_t \) such that \( n_t \sim \mathcal{N}(0_3, C) \) is a 3D noise vector where \( 0_3 \) is a 3D null vector, and \( C = \begin{pmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{pmatrix} \) is the covariance matrix. The per-point dynamic model follows from Equation (3) such that

\[
p_t = p_{t-1} + w_t,
\]

where \( w_t \) is the noisy version of the innovation. We propose to treat each 3D point \( p_t \) in motion as an independent dynamic system decorrelated from other 3D points in the scene. The state \( s_t \) of this dynamic system is defined by the position \( p_t = (x_t, y_t, z_t)^T \) and the velocity \( v_t = (v(x,t), v(y,t), v(z,t))^T \) of the corresponding 3D point such that \( s_t = (x_t, v(x,t), y_t, v(y,t), z_t, v(z,t))^T \). We propose to use the per-point correspondence together with the measurement and dynamic models, and their corresponding measurement and motion uncertainties, to update and filter the system state using a Kalman filter [35].
Following from Equation (5), the measurement model for state $s_t$ is defined as

$$\hat{p}_t = B s_t + n_t, \text{ where } B = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}. \quad (7)$$

In this article, we assume a constant velocity model, where the acceleration $a_t$ of the point $p_t$ is a random vector such that $a_t \sim N(0_3, C_a)$ where $C_a = \begin{pmatrix} \sigma_{ax}^2 & 0 & 0 \\ 0 & \sigma_{ay}^2 & 0 \\ 0 & 0 & \sigma_{az}^2 \end{pmatrix}$. Considering a timestep $\Delta t$ the dynamic model in Equation (6) can be written as

$$p_t = p_{t-1} + v_{t-1} \Delta t + \frac{1}{2} a_t \Delta t^2, \quad (8)$$

and the corresponding velocity is

$$v_t = v_{t-1} + a_t \Delta t,$$  \quad (9)

which can, in turn, be written in the following matrix form:

$$s_t = D s_t + \alpha_t, \text{ such that } D = \begin{pmatrix} D_x & 0 & 0 & 0 & 0 \\ 0 & D_y & 0 & 0 & 0 \\ 0 & 0 & D_z & 0 & 0 \end{pmatrix},$$

where $D_x = D_y = D_z = \begin{pmatrix} 1 & \Delta t \\ 0 & 1 \end{pmatrix}$. Moreover, $\alpha_t$ represents the process error, such that $\alpha_t \sim N(0, Q)$ where $0$ is a 6D null vector and $Q = \begin{pmatrix} \sigma_{ax}^2 & 0 & 0 \\ 0 & \sigma_{ay}^2 & 0 \\ 0 & 0 & \sigma_{az}^2 \end{pmatrix}$, where $A = \begin{pmatrix} \Delta t^2 / 4 & \Delta t / 2 \\ \Delta t / 2 & 1 \end{pmatrix}$.

Now using the standard Kalman equations, the prediction of the next state is given as

$$\begin{cases} \hat{s}_{t|t-1} = D s_{t-1|t-1}, \\ \hat{P}_{t|t-1} = D P_{t-1|t-1} D^T + Q, \end{cases} \quad (11)$$

where $P_{t-1|t-1}$ is the covariance matrix corresponding to the previous state $s_{t-1|t-1}$ and $\hat{P}_{t|t-1}$ is the covariance matrix corresponding to the predicted state $\hat{s}_{t|t-1}$. The error in the predicted state $\hat{s}_{t|t-1}$ is corrected by comparing it with the observed measurement $\tilde{p}_t$ based on the Kalman gain matrix $G_{t|t}$, which is computed as follows:

$$G_{t|t} = \hat{P}_{t|t-1} B^T (B \hat{P}_{t|t-1} B^T + C)^{-1},$$

using this gain $G_{t|t}$, the corrected state vector and covariance matrix are obtained via

$$\begin{cases} s_{t|t} = \hat{s}_{t|t-1} + G_{t|t}(\tilde{p}_t - B \hat{s}_{t|t-1}), \\ P_{t|t} = \hat{P}_{t|t-1} - G_{t|t} B \hat{P}_{t|t-1}. \end{cases} \quad (12)$$

This per-point filtering is performed for each $\tilde{p}_t$ to obtain the filtered, but blurred, estimate of $H_t$, i.e., $\hat{H}_t$. Similarly, we get the filtered 3D velocity estimates for all points, i.e., $\hat{V}_t$, where $\hat{V}_t$ contains $U$ velocity vectors. The proposed method depends on an accurate non-rigid registration step to establish point-to-point correspondences for tracking and refinement. To handle incorrect correspondences, we use a threshold distance parameter $\tau$, which allows for resetting the tracking of points [34]. For each correspondence between points $\tilde{p}_t$ and $\hat{p}_{t-1}$, if $||\tilde{p}_t - \hat{p}_{t-1}|| > \tau$, then the state (and corresponding covariance) of $\hat{p}_{t-1}$ is reset and tracking starts afresh. A filtered/accurate state for such a point is recovered after continuous tracking for a few frames. In the same vein, it is also interesting to discuss robustness of the proposed method in the face of changing or unstable
camera views during a sequence. If there is no traumatic view change, the method should work fine as long as the points are being correctly registered/tracked via the non-rigid registration algorithm as discussed above. A traumatic view change, on the other hand, would mean loss of previous information in the worst case, as tracking for all, or most of, the points would be lost.

It is to be noted that since the measurement noise and the process noise affect each coordinate of the 3D point independently, the per-point Kalman filtering can be split into per coordinate Kalman filtering. This decreases the complexity of computation of the Kalman gain matrix \( G_{t|t} \) for each point. In the case of commodity depth cameras, the 3D point measurements suffer from depth-dependant measurement noise instead of per-coordinate independent noise as discussed above [9, 41]. The details of depth dependant measurement noise model together with its affects on the noise covariance matrix \( C \) are discussed in the Appendix A.1.

### 4.3 Proposed 3D BTV Deblurring

Per-point refinement via tracking discussed in Section 4.2, although allowing for view-independence, does not explicitly cater for blurring in the measurement model in Equation (2) [51]. Furthermore, blurring artifacts are introduced due to treating each point separately, which affects the global smoothness property of point clouds [34]. This results in filtered but blurred estimates of 3D point positions in \( \mathcal{H}_t \), i.e., \( \hat{\mathcal{H}}_t \), together with the corresponding velocity estimates, i.e., \( \hat{\mathbf{V}}_t \). Therefore, after per-point tracking, at every timestep, it is necessary to carry out deblurring and regularization of position and motion estimates at hand to produce deblurred and globally smooth estimates [34]. We carry out the 3D BTV regularization of position estimates via the following minimization framework:

\[
\hat{\mathcal{H}}_t = \arg \min_{\mathcal{H}_t} \mu |\nabla \mathcal{H}_t| + \frac{1}{2} \| \mathcal{H}_t - \hat{\mathcal{H}}_t \|^2_2,
\]  

which defines an \( L_2 \)-optimization with an \( L_1 \)-BTV regularization \( |\nabla \mathcal{H}_t| \). \( \nabla \mathcal{H}_t \) represents the discrete gradient of \( \mathcal{H}_t \), \( |.| \) denotes the L1-norm and \( \mu \) is the regularization parameter. BTV regularization/denoising has been a topic of interest for researchers but most of the research has been restricted to organized color and depth images [28, 33, 36, 39, 42], where the neighborhoods are well defined and the gradient, based on intensity or depth values, is easy to compute, e.g., via shift operators [34, 51]. In the current problem, \( \hat{\mathcal{H}}_t \) is a set of unorganized 3D points without any connectivity or neighborhood information, therefore the extension of BTV regularization to 3D point clouds is not a straightforward problem. We are interested in finding a gradient operator \( \nabla \), which computes gradient per 3D point by taking into account the properties of the underlying surface in its local neighborhood. Therefore, we choose \( \nabla \) such that it exploits the properties of local point patches based on their unique locations, geometry, and curvature, as illustrated in Figure 2, to formulate the 3D BTV regularization such that

\[
|\nabla \mathcal{H}_t| = \sum_{i,j} \| \nabla p_{ij}^t \|
\]

\[
= \sum_{p_{i}^t, p_{j}^t \in \Omega_t^c} \frac{w_{ij}^t w_{ij}^t (p_{i}^t - p_{j}^t) - (p_{i}^t - p_{j}^t)) \|}{w_{ij}^t (p_{i}^t, p_{j}^t)}
\]

where \( \Omega_t^c \) is the pre-computed neighborhood of \( p_{i}^t \) and \( \omega_t^c = \sum_{p_{i}^t, p_{j}^t \in \Omega_t^c} w_{ij}^t \). Each local patch corresponding to the neighborhood \( \Omega_t^c \) of the query point \( p_{i}^t \) is characterized by the mean and covariance, i.e., \( (p_{i}^t, C_{i}^t) \), of the points in it. Similarly the patch corresponding to \( p_{j}^t \) is characterized by \( (p_{j}^t, C_{j}^t) \). We assume equal distribution of points in \( \hat{\mathcal{H}}_t \), therefore we have \( C_{i}^t = C_{j}^t \).
Fig. 2. Illustration of main components for per-point gradient computation on a 2D surface. $p^i_t$ is the query point and $p^j_t$ lies in its neighborhood. Their corresponding neighborhoods are represented by $\Omega^i_t$ and $\Omega^j_t$, and the local point patches corresponding to these neighborhoods are classified by the mean and covariance of points in them, i.e., $(\bar{p}^i_t, C^i_t)$ and $(\bar{p}^j_t, C^j_t)$, respectively. $c^{ij}_t$ is the Euclidean distance between $p^i_t$ and $p^j_t$, and $d^{ij}_t$ is the shortest distance of $p^j_t$ to the plane, tangent at $p^i_t$ to the local patch of $p^i_t$, defined via the normal vector $\vec{u}^i_t$.

sizes of pre-computed neighborhood depend on the noise level in the data. The objective is to have a neighborhood large enough that can be used to compute the properties of the underlying patch as accurately as possible. On the other hand, larger neighborhoods result in an increase in computation complexity.

Now we localize $p^i_t$ and $p^j_t$ by subtracting from them the corresponding means and then find the difference between their local positions. This difference is then weighted by two factors $w^{ij}_{(t,c)}$ and $w^{ij}_{(t,d)}$, where $w^{ij}_{(t,c)}$ is defined as

$$w^{ij}_{(t,c)} = \exp(-(c^{ij}_t)^2/2\sigma^2_c),$$

where $c^{ij}_t = \|p^i_t - p^j_t\|$ is the Euclidean distance between $p^i_t$ and $p^j_t$, and $\sigma^2_c$ is a constant thresholding factor. The weight $w^{ij}_{(t,c)}$ serves to give more importance to points that lie closer to $p^i_t$. On the other hand, $w^{ij}_{(t,d)}$ is defined as

$$w^{ij}_{(t,d)} = \exp(-(d^{ij}_t)^2/2\sigma^2_d),$$

where $d^{ij}_t = (\vec{u}^i_t)^T(p^i_t - p^j_t)$ is the shortest distance of $p^j_t$ to the plane tangent, at $p^i_t$, to the underlying surface sampled by the local patch of $p^i_t$. The vector $\vec{u}^i_t$ is the normal vector to the plane at $p^i_t$, and $\sigma^2_d$ is a constant thresholding factor. $w^{ij}_{(t,d)}$ also serves to detect outliers and to preserve the edge information by taking into account the change in curvature in the local patch of $p^i_t$.

The $L_2$-norm in Equation (14) is convex and differentiable whereas the $L_1$-norm is convex and non-differentiable (non-smooth). Such type of problems cannot be solved by using simple gradient-decent methods [28]. Therefore, we use the forward-backward splitting (FBS) method (also known as proximal gradient solver), which relies on computing a proximal operator for the non-smooth part of the problem, which is implemented using fast adaptive shrinkage/thresholding algorithm (FASTA) [28]. $|\nabla H_t|$ is first reformulated to a simpler form, which is differentiable, by defining a vector $r^{ij}_t \in \mathbb{R}^3$ and using Cauchy-Swartz inequality to write [28]

$$\max_{\|r^{ij}_t\| \leq 1} \langle r^{ij}_t, \nabla p^{ij}_t \rangle = \|\nabla p^{ij}_t\|,$$
where \( r_{ij}^t \) is assumed to be parallel to \( \nabla p_{ij}^t \), having a unit norm. Using this definition of \( \| \nabla p_{ij}^t \| \) in Equations (14) and (15), respectively, solving Equation (14) is equivalent to finding

\[
\max_{\| r_{ij}^t \| \leq 1} \arg \min_{\mathcal{H}_t} \mu \langle \mathcal{R}_t, \nabla \mathcal{H}_t \rangle + \frac{1}{2} \| \mathcal{H}_t - \mathcal{H}_t^f \|_2^2. \tag{19}
\]

where \( \mathcal{R}_t = \{ r_{ij}^t \} \), and the inner minimization is now differentiable. The minimal value of \( \mathcal{H}_t \) for a given value of \( \mathcal{R}_t \) should satisfy

\[
\mathcal{H}_t = \mathcal{H}_t^f + \frac{\mu}{2} \nabla \cdot \mathcal{R}_t,
\]

where \( \nabla \cdot \) is the discrete divergence operator and can be computed by taking transpose of the gradient operator. We can reformulate Equation (19) using the optimal value of \( \mathcal{H}_t \) to get dual form of Equation (14) such that

\[
\hat{\mathcal{R}}_t = \arg \min_{\| \mathcal{R}_t \|_\infty \leq 1} \frac{1}{2} \| \nabla \cdot \mathcal{R}_t - \frac{1}{\mu} \mathcal{H}_t^f \|_2^2.
\tag{20}
\]

This problem is solved via the FBS method as explained in Reference [28], and the final deblurred result at time \( t \) is obtained via

\[
\hat{\mathcal{H}}_t = \mathcal{H}_t^f + \mu \nabla \cdot \hat{\mathcal{R}}_t.
\tag{21}
\]

In the case \( \nabla \) is linear, it can be represented as a sparse matrix for which the corresponding discrete divergence operator can be computed by taking the transpose of this sparse matrix. This makes the solution of this problem very efficient. Therefore, for making \( \mathcal{H}_t^f \) linear, we use the input \( \hat{\mathcal{H}}_t^f \) to pre-compute the neighborhoods \( \Omega_i^t \) and \( \Omega_j^t \), the weights \( w_{ij}(t,c) \) and \( w_{ij}(t,d) \), and the normals \( \hat{\mathcal{N}}_j^t \), for all points. This method, although effective, is sensitive to parameters and can result in over-smoothing of the output. Therefore, similar to the work done in the image domain [34, 36, 39, 42], we propose to use iterative regularization with the minimization in Equation (14) carried out multiple times, whereby in each iteration the regularization parameter \( \mu \) is decreased in a dyadic way. This produces enhanced and feature-preserving point clouds as shown in the results.

In the next step, we want to use the deblurred point cloud \( \hat{\mathcal{H}}_t \) to correct the per-point constant velocities estimate \( \hat{\mathcal{V}}_t^f \). For this purpose, we use \( \hat{\mathcal{H}}_t \) and the previous result \( \hat{\mathcal{H}}_{t-1} \) to compute the per-point corrected velocities estimate \( \hat{\mathcal{V}}_t^i \in \mathbb{R}^3 \) via

\[
\hat{\mathcal{V}}_t^i = \frac{(\hat{\mathcal{P}}_t^i - \hat{\mathcal{P}}_{t-1}^i)}{\Delta t},
\tag{22}
\]

to get \( \hat{\mathcal{V}}_t = \{ \hat{\mathcal{V}}_t^i \} \). These corrected velocity estimates are then used to produce the per-point corrected state estimates, which are then used in the next iteration.

5 EXPERIMENTS AND RESULTS

In this section, we present the results of the quantitative and qualitative analysis of performance of the proposed recursive dynamic 3D SR method using both synthetic and real experimental data. The data is in the form of 3D videos and contains non-rigid objects undergoing local motions of various complexities. We start by analyzing the results of our experiments on synthetic data, which includes evaluation of different steps of the proposed method and its comparison with the state-of-the-art methods. This is followed by an analysis of results of the proposed method using real data acquired by cameras in a multi-view system. We show the ability of the proposed 3D SR method to enhance LR and noisy 3D reconstructions of non-rigid objects undergoing local motions as well as significant topology changes.

5.1 Evaluation on Synthetic Data

In this section, we analyze the performance of the proposed method, using synthetic data with available GT, both qualitatively and quantitatively. This performance analysis includes analyzing
The effects of different steps of the proposed pipeline followed by a comparison with the state-of-the-art filtering methods under varying noise and SR levels.

We use the “Samba” dataset [58], which contains high quality meshes from which HR 3D point clouds are extracted. This HR data represents full 3D reconstructions of real scenes of a non-rigid human body, undergoing smooth and non-smooth local motions over time as shown in Figure 8, which we call GT. We use 35 frames from this sequence for our experiments.

We start by analyzing the effects of different steps of the proposed SR pipeline as shown in Figure 1. For this purpose, the GT point clouds are first downsampled by a SR factor $\alpha = 4$, then zero-mean Gaussian noise is added independently to each coordinate of 3D points, of the downsampled GT clouds, with standard deviations $\sigma_x = \sigma_y = \sigma_z = 3\text{cm}$. These LR noisy point clouds are given as input and SR results of upsampling based on mesh edge division using GT mesh information with $\alpha = 4$, upsampling and per-point tracking using a Kalman filter, and upsampling, per-point tracking together with multi-level iterative 3D BTV deblurring, are obtained. Root-mean-squared error (RMSE) for the result of each method is computed with respect to the HR GT data. Figure 3 shows the RMSE per frame for each of the steps mentioned before. Although per-point tracking using a Kalman filter recursively enhances the 3D point clouds and requires only three to four frames to converge, its performance is limited by its inability to handle system blur and its ability to introduce noisy artifacts. Adding a deblurring step based on 3D BTV regularization ($\sigma_c = 1.8\text{cm}, \sigma_h = 1.65\text{cm}$) solves this problem and produces the best results.

In the next experiment, we perform a comparison of the state-of-the-art static 3D point cloud enhancement methods with the proposed dynamic SR scheme using the data affected by noise of varying magnitude. The GT point clouds are downsampled and upsampled by a factor $\alpha = 4$ as explained above. Zero-mean Gaussian noise of standard deviations $\sigma_x = \sigma_y = \sigma_z = 1\text{cm}, 2\text{cm},$ and $3\text{cm}$, is added to the downsampled GT point clouds, respectively. In addition to the proposed method, we use static filtering schemes based on bilateral mesh denoising (BMD) [25] and MLS [4] to enhance the upsampled point clouds. RMSE per frame for results of BMD, MLS, proposed method with per-point tracking only, and proposed method with per-point tracking and 3D BTV deblurring are plotted in Figure 4. Although the proposed method, with per-point tracking only, is able converge more quickly as the noise level decreases, its performance remains worse than the
Fig. 4. Comparison of the proposed technique with the state-of-the-art methods for enhancement of 3D measurements, corresponding to non-rigid objects, affected by noise of varying magnitude. 35 LR frames of the “Samba” dataset [58], with zero-mean Gaussian noise of standard deviations 1cm, 2cm, and 3cm added to each coordinate of 3D points independently are used, respectively. The SR factor is $\sigma = 4$. Two static filtering methods namely bilateral mesh denoising (BMD) [25] and moving least squares (MLS) [4] are compared with the proposed recursive and dynamic SR method with (UPTrackTV) and without (UPTrack) the 3D BTV deblurring. BMD1 is the result of BMD on data affected by Gaussian noise of standard deviation 1cm, and so on. Results show that UPTrackTV provides the best performance, as compared to the other methods, across all noise levels with its comparative performance improvement increasing with increasing data noise. This is due to its ability to tackle noisy artifacts locally as well as globally, in contrast with other methods, which are mainly local in nature and hence, are unable to tackle high magnitude of noise in the data.

other methods due to introduction of blurring artifacts. The performance of BMD and MLS starts to get worse with the increase in noise magnitude due to their local nature and their inability to handle highly noisy artifacts. The proposed method with per-point tracking and 3D BTV blurring provides the best performance at all noise levels and can produce globally smooth and feature-preserving point clouds even at high noise levels. The $(\sigma_c, \sigma_h)$ values used for these experiments with Gaussian noise $\sigma_x = \sigma_y = \sigma_z = 1$cm, 2cm, and 3cm are $(1.1$cm, 0.65$cm)$, $(1.5$cm, 1.3$cm)$, and $(1.8$cm, 1.65$cm)$, respectively.

In Figure 7, we plot mesh reconstructions of an example frame (#33), which are obtained as a result of adding independent Gaussian noise to each coordinate of the downsampled GT data with standard deviation of 1cm, dense upsampling of LR noisy data with $\sigma = 4$ only, upsampling and BMD, upsampling and MLS, proposed pipeline with $\sigma = 4$, together with HR GT meshes. The meshing of point clouds is carried out by using the mesh information available for GT. The results clearly show that the proposed technique produces enhanced, smoother, and feature-preserving reconstruction as compared to other methods. BMD and MLS fail to preserve smaller features such as hands, arm, nose, and the like. To investigate further the quality of reconstructions obtained via the methods mentioned above, we calculate the RMSE for different body parts for the reconstructed example frame#33. Table 1 shows these results, from which it is clear that even for separate body parts the conclusions drawn above hold.

Figure 8 shows plots of 3D mesh reconstruction of five frames (#1, #3, #12, #21, #30) from the sequence, obtained as a result of the proposed method. It shows that the proposed method is able to
Fig. 5. Comparison of the proposed technique with the state-of-the-art methods for 3D point cloud enhancement for different SR factors. 35 LR frames (downsampled by a factor $o = 4$) of the “Samba” dataset [58], with zero-mean Gaussian noise of standard deviation 3cm added to each coordinate of 3D points independently are used. The filtering is performed on the input data upsampled by a factor $o = 1$ and $o = 4$, respectively. Two static filtering methods, namely BMD [25] and MLS [4], are compared with the proposed recursive and dynamic SR method. BMD1 is the result of BMD on input LR and noisy data upsampled by a factor $o = 1$, and so on. Although the proposed method has comparative performance at $o = 1$ with respect to the performance of the state-of-the-art methods at $o = 4$, it achieves best results at $o = 4$.

Fig. 6. Comparison of the proposed technique with the state-of-the-art SR methods, namely conventional bicubic interpolation, recUP-SR [34], and SISR [6], for enhancement of 3D/depth videos generated by simulating a mono-view depth system using the “Samba” dataset [38]. Nineteen LR depth frames with zero-mean Gaussian noise of standard deviation 3cm added to the depth measurements are used [34]. The results show improved accuracy of the proposed method as compared to the other methods.

Table 1. 3D RMSE in mm for Different Body Parts, of Frame#33 of the “Samba” Dataset [58], using Different Methods as Shown in Figure 7

|        | Arm | Leg | Torso | Full body |
|--------|-----|-----|-------|-----------|
| LR     | 11.31 | 11.61 | 11.03 | 11.48     |
| UP     | 9.43  | 10.23 | 9.55  | 9.84      |
| BMD    | 9.22  | 9.03  | 7.46  | 8.23      |
| MLS    | 10.07 | 8.83  | 7.75  | 8.69      |
| Proposed | **8.05** | **7.55** | **7.26** | **7.83** |
Fig. 7. 3D mesh plots of a super-resolved resultant frame #33 from the “Samba” dataset [58] after (b) dense upsampling (UP), (c) bilateral mesh denoising (BMD), (d) moving least squares (MLS), and (e) proposed recursive and dynamic SR scheme. (a) is the 3D plot of LR noisy data and (f) is the GT HR mesh respectively. Proposed technique produces smooth, enhanced and feature preserving reconstruction as compared to the rest. The SR factor is $\sigma = 4$. Display color-scale is based on mean surface curvature.

Proposed technique recursively enhances the noisy input measurements while successfully tackling non-rigid smooth and non-smooth local motions.

In the next experiment, we perform a comparison of the state-of-the-art static 3D point cloud enhancement methods, i.e., BMD and MLS, with the proposed dynamic SR scheme for different SR factors. This means that GT point clouds are first downsampled by a SR factor $\sigma = 4$, then zero-mean Gaussian noise is added independently to each coordinate of 3D points, of the downsampled GT clouds, with standard deviations $\sigma_x = \sigma_y = \sigma_z = 3\text{cm}$. Filtering is carried on this data with upsampling factors of $\sigma = 1$ and $\sigma = 4$, respectively. RMSE per frame is plotted in Figure 5. Results show that the proposed method clearly outperforms both BMD and MLS when used on the same data. The results also show that even at upsampling factor $\sigma = 1$, the proposed dynamic scheme gives comparative performance with respect to both BMD and MLS used on upsampled noisy data with $\sigma = 4$. This is outperformed by applying the proposed dynamic filtering scheme at $\sigma = 4$. The reason for this is that at $\sigma = 1$, the method recursively denoises the noisy input. On the other hand, at $\sigma = 4$, the method applies the full recursive dynamic SR pipeline, which together with denoising, enhances the quality of data by preserving useful features.

Lastly, we perform a comparison of the proposed dynamic 3D SR method with the state-of-the-art SR methods, which include the conventional bicubic interpolation, the dynamic depth SR method proposed by Al Ismaeil et al. [33, 34], called recUP-SR, and the learning-based SR method called SISR proposed by Mac Aodha et al. [6]. We again make use of the Samba dataset [58], and simulate a depth camera, placed at a distance of approximately two meters, in V-Rep [23] to generate a mono-view synthetic depth sequence [34]. This GT depth sequence is downsampled by a factor $\sigma = 4$, and zero mean Gaussian noise of variance $\sigma_z = 3\text{cm}$ is added to the depth measurements. This LR noisy depth sequence is given as input to the state-of-the-art methods, and is converted to a 3D sequence via the known camera parameters and given as input to the proposed method. To compare the super-resolved (by a factor $\sigma = 4$) results of all methods, the resulting depth sequences from state-of-the-art methods and the GT depth sequence are converted to 3D sequences as explained before. After that, per-frame RMSE for the result of each method with respect to the 3D GT is computed. The results are reported in the Figure 6. The results show the robustness and improved accuracy of the proposed method as compared to the state-of-the-art
methods. SISR produces HR 3D reconstructions but its accuracy suffers due to its patch-based nature, which prevents it from preserving finer details. Moreover, both SISR and bicubic interpolation suffer due to not taking into account the temporal information. recUP-SR tries to overcome this weakness by proposing a dynamic and recursive SR scheme, but it is restricted in tracking large and abrupt motions due to working with depth images and range flow motion approximation. In contrast, the results show improved accuracy of the proposed method due to its robustness to handle non-smooth/abrupt motions, undergone by the non-rigid object, which results from its ability to accurately track the positions of points in 3D space. The \((\sigma_c, \sigma_h)\) values for deblurring used in these experiments are (1.125cm, 1cm).

5.2 Evaluation on Real Data

In this section, we analyze the performance of the proposed method using real data acquired via multi-view systems composed of photometric and commodity depth cameras, respectively. In addition to showcasing the ability of the proposed method to enhance the quality of LR and noisy data to produce smooth and feature-preserving full 3D reconstructions of non-rigid objects, this experimental analysis also demonstrates the capabilities of the proposed method to produce accurate and enhanced 3D reconstructions of objects with changing topologies.

In the first experiment, we use full 3D point-clouds extracted from meshes of the “adult child ball” sequence from Inria’s 4D-Repository [30]. This dataset is acquired via a fully calibrated multi-view system based on photometric RGB cameras. The sequence, used here, has two characteristics; the resolution of data is quite low (approximately 10000 points per scene) resulting in non-smooth surfaces, and it contains an object, i.e., a ball, with changing topology as shown in Figure 9. Due to these characteristics, this type of dataset is very challenging for the class of methods to which belong the works by Dou and Fuchs [21, 22], DynamicFusion [45], etc. These methods do not explicitly target LR data and are very sensitive to objects with changing topologies due to their design of always fusing the current measurement with the first frame, which is considered to be the reference. The proposed method, on the other hand, explicitly targets LR 3D data and produces HR, smooth, and feature-preserving 3D reconstructions as shown in Figure 9. Moreover, it works by recursively fusing the current measurement and the result of the previous iteration/timestep and, hence, can accurately reconstruct objects, in this case a ball, with changing topologies. The \((\sigma_c, \sigma_h)\) values for deblurring used in these experiments are (0.5cm, 0.3cm).
Fig. 9. 3D mesh plots of three LR frames (#5, #12, and #15) from Inria’s 4D-Repository [30], i.e., the top row, and the corresponding super-resolved (using SR factor $o = 4$) results of the proposed algorithm, i.e., the bottom row. The input data has low resolution, which results in non-smooth surfaces, thick edges, and loss of details. The results show super-resolved, smooth, and feature-preserving 3D reconstructions of non-rigid objects. They also show ability of the proposed method to produce enhanced reconstructions of objects with changing topologies, e.g., the ball in the above plots. Display color-scale is based on mean surface curvature.

In the next experiment, we use point clouds from the full 3D video of the “jumping in place” action performed by a human subject from the Berkeley Multimodal Human Action Database (MHAD) [47]. This dataset is acquired via a fully calibrated multi-view system composed of two Kinect v1 cameras placed at opposite corners of the acquisition space. As explained in Section A.1, the depth acquisition system of cameras, built on structured-light principle, such as Kinect v1, suffers from depth-dependent measurement noise. The distance of Kinect cameras from the subjects in MHAD’s multi-view setup is approximately 3.5–4 meters. This results in highly noisy 3D measurements with non-smooth surfaces and diminished features as shown in Figure 10. Figure 10 also shows the point clouds which are received as the output of the proposed algorithm. The input data is upsampled by a factor $o = 1.5$. Moreover, to tackle the depth-dependent measurement noise specific to Kinect v1 cameras, the measurement model presented in Section A.1 is used during the per-point tracking step. The resolution enhancement together with per-point tracking and 3D BTV deblurring results in point clouds that are relatively noise-free, have smoother surfaces with less holes/gaps, and better preserved features/details. The $(\sigma_c, \sigma_h)$ values for deblurring used in these experiments are (1.25cm, 0.75cm).

Lastly, it is to be noted that we have used different upsampling operators to test the proposed approach. These include the mesh sub-division operator for datasets with available mesh information, e.g., full 3D Samba dataset [58] (Section 5.1 and Figure 3). We have also used the bi-linear
Fig. 10. Plots of LR 3D point-clouds of five frames (#6, #10, #18, #22, and #35) from the Berkeley MHAD Kinect dataset [47], on the left, and the corresponding super-resolved (using SR factor $o = 1.5$) results of the proposed algorithm on the right. The input data suffers from high magnitude of noisy artifacts, in the form of non-smooth surface and jagged edges, due to large distance of the human subject from the cameras. The results show super-resolved, smooth, and feature preserving full 3D reconstructions of the human subject.

Table 2. Computation Times (sec) for Different Stages of the Proposed Scheme for Each Frame of the Samba Dataset [58], as Shown in Figure 7, and the Berkley MHAD Dataset [47], as Shown in Figure 10

|                  | Points/Frame | Registration | Tracking | Deblurring |
|------------------|--------------|--------------|----------|------------|
| Samba            | 10,000       | 120 sec      | 3.5 sec  | 40 sec     |
| Berkley MHAD     | 54,000       | 1500 sec     | 16 sec   | 350 sec    |

interpolation operator for datasets based on depth cameras, e.g., Berkley MHAD dataset [47] (Figure 10). The proposed approach has been shown to be robust to both types of upsampling operators.

5.3 Performance & Implementation Details
In this section, we report the runtime performance of the proposed scheme on the datasets we evaluated in the previous sections. The implementation of the proposed scheme together with the experimental evaluation has been carried out in Ubuntu 14.04 on a desktop system with Intel Xeon 3.4GHz (8 cores) processor and 8GB of RAM.

The proposed scheme can be divided into three main steps, first is the non-rigid registration based on CPD algorithm, second is per-point tracking and refinement based on Kalman filter, and third is deblurring based on 3D BTV. For the non-rigid registration, we have used the standard CPD implementation provided by the authors [44]. Similarly, for 3D BTV deblurring we use the FASTA implementation of the FBS method provided by the authors [28]. The per-point refinement via Kalman filter has been implemented in C++.

We take the full 3D sequence from the Samba dataset [58], as shown in Figure 7, and the Berkley MHAD dataset [47], as shown in Figure 10, as examples. Each frame of Samba dataset contains approximately 10,000 3D points whereas each frame of the Berkley MHAD dataset contains approx. 54,000 3D points. The computation-times for the three stages for processing one frame of these datasets are given in Table 2. This table shows that the maximum time is required by the CPD based non-rigid registration algorithm followed by the 3D BTV deblurring and per-point tracking, respectively. It is to be noted that the tracking implementation is not optimized for tackling each
point independently and in parallel fashion. Therefore, we believe that we can achieve real-time performance for tracking with better implementation. Lastly, around 80% of the computation time during deblurring is used in computation of the matrices related to the gradient operator. This operation can also be highly optimized via parallelization, e.g., with the help of a GPU processor.

6 CONCLUSION & FUTURE WORK

In this article, we have presented a framework for acquiring high quality and full 360° 3D reconstructions of dynamic scenes containing non-rigid objects undergoing large local motions/deformations. We target noisy and LR data acquired from commodity 3D cameras in mono-view or multi-view systems. This framework is based on a view-independent recursive and dynamic multi-frame 3D SR algorithm, which is capable of filtering out the noise as well as enhancing the resolution of the raw measurements obtained from multi-view systems. The proposed algorithm tracks and filters the position and motion of every 3D point recursively, hence making use of complete 3D characteristics of the input data. It is able to handle generic 3D as well as structured-light, sensing-based, depth-specific noise in 3D measurements. Moreover, it uses a 3D BTV regularization for deblurring and smoothing of the point clouds after per-point tracking. Quantitative and qualitative evaluation of the proposed framework shows its better performance as compared to state-of-the-art methods for producing noise-free and smooth full 3D reconstructions.

As future work, we would be interested in incorporating the proposed 3D SR scheme to improve the accuracy of applications based on extracting meaningful information from 3D measurements. A facial recognition system is an example of such an application that could benefit from the high-quality 3D reconstructions obtained from the proposed scheme in un-constrained environments [7, 11]. We would also like to investigate the use of texture information, which is available via RGB sensors in commodity RGB-D cameras [9, 41]. This information can be used to produce high-quality textured 3D reconstructions [21]. Furthermore, its utility in improving the performance of the proposed 3D BTV framework can also be investigated [62]. Lastly, we would also like to investigate the use of a curvature operator instead of a gradient operator in the proposed 3D BTV framework [48].

A APPENDIX

A.1 Depth Dependent Measurement Noise

The measurement model in Equation (5) assumes per-coordinate independent Gaussian noise affecting each 3D point \( p_t \). In reality, the 3D points are computed from depth images acquired via commodity 3D cameras built on structured-light or time-of-flight principles [9, 41, 49]. The acquired per-point depth measurement, i.e., \( \hat{q}_t \) is defined by the approximated pixel position \((\tilde{u}_t, \tilde{v}_t, \tilde{z}_t)\) in the depth image, and the measured depth value \( \tilde{z}_t \) such that \( \hat{q}_t = q_t + n_t \), where \( n_t = (n_{(u,t)}, n_{(v,t)}, n_{(z,t)})^T \) represents noise in the measured pixel position and depth value. Let us consider a structured-light depth camera [9], for which the depth measurement \( \tilde{z}_t \) suffers due to noise \( n_{(d,t)} \) in disparity \( d \), which is the distance (in pixels) between locations of a point in an observed and projected pattern, via the relation \( n_{(z,t)} = -\frac{z_t^2}{f^2} n_{(d,t)} \), where \( f \) is camera’s horizontal focal length, \( b \) the baseline distance between the camera and the projector, and \( n_{(d,t)} \) is the noise in the corresponding disparity measurement \( d_t \) [2, 55]. The main factor affecting both the pixel and disparity measurements is the noise due to quantization [2], therefore we can assume it to be drawn from independent Gaussian distributions such that \( n_{(u,t)} \sim N(0, \sigma^2_u) \), \( n_{(v,t)} \sim N(0, \sigma^2_v) \) and \( n_{(d,t)} \sim N(0, \sigma^2_d) \). This allows us to model the noise in depth measurement, i.e., \( n_{(z,t)} \sim N(0, \sigma^2_{(z,t)}) \) where \( \sigma^2_{(z,t)} = \left( \frac{z_t^2}{f^2 b} \right)^2 \sigma^2_d \).
To convert the depth measurement $\tilde{q}_t$ to the corresponding 3D position $\tilde{p}_t$, the intrinsic matrix $K = \begin{pmatrix} f_u & 0 & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{pmatrix}$, where $(f_u, f_v)$ represent the focal lengths (where $f = f_u$), and $(c_u, c_v)$ represent center of camera’s imager such that

$$\tilde{p}_t = \tilde{Z}_t K^{-1} \tilde{q}_t = \tilde{Z}_t K^{-1}(q_t + \tilde{n}_t),$$

(23)

where $\tilde{Z}_t = \begin{pmatrix} \tilde{z}_t & 0 & 0 \\ 0 & \tilde{z}_t & 0 \\ 0 & 0 & 1 \end{pmatrix}$ and $\tilde{z} = z + n_{(z,t)}$. Therefore, the measurement model for each 3D point can now be defined as

$$\tilde{p}_t = p_t + n'_t,$$

(24)

where

$$n'_t = \begin{pmatrix} z_t n_{(u,t)} + (u_t - c_u) n_{(u,t)} + n_{(u,t)} n_{(u,t)} \\ \tilde{z}_t n_{(v,t)} + (v_t - c_v) n_{(v,t)} + n_{(v,t)} n_{(v,t)} \\ n_{(z,t)} \end{pmatrix}^T.$$

(25)

Here $n'_t \sim \mathcal{N}(0_3, C'_t)$, where the entries of covariance matrix $C'_t$ are defined as

$$\begin{align*}
\text{cov}(n_{(x,t)}, n_{(x,t)}) &= \frac{\sigma_x^2 + (u_t - c_u)^2 \sigma_{(x,t)}^2}{f_u}, \\
\text{cov}(n_{(y,t)}, n_{(y,t)}) &= \frac{\sigma_y^2 + (v_t - c_v)^2 \sigma_{(y,t)}^2}{f_v}, \\
\text{cov}(n_{(z,t)}, n_{(z,t)}) &= \frac{(u_t - c_u)^2 \sigma_{(z,t)}^2}{f_u}, \\
\text{cov}(n_{(x,t)}, n_{(y,t)}) &= \frac{(u_t - c_u)(v_t - c_v) \sigma_{(z,t)}^2}{f_u f_v}, \\
\text{cov}(n_{(x,t)}, n_{(z,t)}) &= \frac{(u_t - c_u) \sigma_{(z,t)}^2}{f_u}, \\
\text{cov}(n_{(y,t)}, n_{(z,t)}) &= \frac{(v_t - c_v) \sigma_{(z,t)}^2}{f_v},
\end{align*}$$

(26)

where $\text{cov}(\ldots)$ computes the covariance between two random variables. This covariance matrix, specific to each point, can therefore be replaced in Equation (12) when dealing with data acquired from depth cameras. To compute this covariance matrix, the noise-free pixel and depth values are required, but are not available in practice. Therefore, we propose to use the measured pixel and depth values instead, which are the closest approximation of the noise-free values we can get. Using the $C'_t$ increases complexity of the proposed approach, as now we have to deploy a Kalman filter per point, instead of per coordinate, which was the case previously, but it captures the noise characteristics of depth cameras more accurately.

REFERENCES

[1] Hassan Afzal, Djamila Aouada, François Destelle, Bruno Mirbach, and Björn Ottersten. 2015. View-independent enhanced 3D reconstruction of non-rigidly deforming objects. In Proceedings of the 16th International Conference on Computer Analysis of Images and Patterns (CAIP’15), Part II. Springer, 712–724. DOI: http://dx.doi.org/10.1007/978-3-319-23117-4_61

[2] H. Afzal, D. Aouada, D. Fofi, B. Mirbach, and B. Ottersten. 2014. RGB-D multi-view system calibration for full 3D scene reconstruction. In Proceedings of the 2014 22nd International Conference on Pattern Recognition (ICPR’14). IEEE, 2459–2464. DOI: http://dx.doi.org/10.1109/ICPR.2014.425

[3] H. Afzal, K. Al Ismaiel, D. Aouada, F. Destelle, B. Mirbach, and B. Ottersten. 2014. KinectDeform: Enhanced 3D reconstruction of non-rigidly deforming objects. In Proceedings of the 3DV Workshop on Dynamic Shape Measurement and Analysis, Vol. 2. IEEE, 7–13. DOI: https://doi.org/10.1109/3DV.2014.114

[4] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and Claudio T. Silva. 2003. Computing and rendering point set surfaces. IEEE Trans. Vis. Comput. Graph. 9, 1 (Jan. 2003), 3–15. DOI: http://dx.doi.org/10.1109/TVCG.2003.1175093

[5] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. 2005. SCAPE: Shape completion and animation of people. ACM Trans. Graph. 24, 3 (July 2005), 408–416. DOI: http://dx.doi.org/10.1145/1073204.1073207

[6] Oisin Mac Aodha, Neill D. F. Campbell, Arun Nair, and Gabriel J. Brostow. 2012. Patch based synthesis for single depth image super-resolution. In Proceedings of the 12th European Conference on Computer Vision (ECCV’12), Vol. Part III. Springer-Verlag, Berlin, 71–84. DOI: http://dx.doi.org/10.1007/978-3-642-33712-3_6

[7] Djamila Aouada, Kassem Al Ismaiel, Kedija Kedir Idris, and Björn E. Ottersten. 2014. Surface UP-SR for an improved face recognition using low resolution depth cameras. In Proc. 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS’14). IEEE, 107–112. DOI: http://dx.doi.org/10.1109/AVSS.2014.6918652

ACM Trans. Multimedia Comput. Commun. Appl., Vol. 14, No. 1s, Article 24. Publication date: March 2018.
