View-Independent Enhanced 3D Reconstruction of Non-Rigidly Deforming Objects

Hassan Afzal[1], Djamila Aouada[1], François Destelle[2], Bruno Mirbach[3], and Björn Ottersten[1]

[1] Interdisciplinary Centre for Security, Reliability and Trust
University of Luxembourg, 4, rue Alphonse Weicker, L-2721, Luxembourg
Email: {hassan.afzal, djamila.aouada, bjorn.ottersten}@uni.lu
[2] Dublin City University, Insight: Centre for Data Analytics, Ireland
Email: francois.destelle@dcu.ie
[3] IEE S.A., Advanced Engineering, Contern, Luxembourg
Email: bruno.mirbach@iee.lu

Abstract. In this paper, we target enhanced 3D reconstruction of non-rigidly deforming objects based on a view-independent surface representation with an automated recursive filtering scheme. This work improves upon the *KinectDeform* algorithm which we recently proposed. *KinectDeform* uses an implicit view-dependent volumetric *truncated signed distance function (TSDF)* based surface representation. The view-dependence makes its pipeline complex by requiring surface prediction and extraction steps based on camera's field of view. This paper proposes to use an explicit projection-based *Moving Least Squares (MLS)* surface representation from point-sets. Moreover, the empirical weighted filtering scheme in *KinectDeform* is replaced by an automated fusion scheme based on a *Kalman filter*. We analyze the performance of the proposed algorithm both qualitatively and quantitatively and show that it is able to produce enhanced and feature preserving 3D reconstructions.

1 Introduction

Data acquired by commodity 3D sensing technologies is noisy and of limited resolution. This limits its direct use in applications ranging from environment mapping for mobile autonomous systems and preservation of historical sites, to human activity and gesture recognition for virtual communications, assistive robotics, security and surveillance.

Research has been carried out to build techniques around commodity 3D sensing technologies to accurately reconstruct captured 3D objects or scenes by relying on training data or use of templates such as in the case of [13] and [25] or by fusing a specified number of captured frames to produce a single high quality 3D reconstruction [19]. *KinectFusion* and similar techniques provide an effective and efficient mechanism to recursively fuse and filter the incoming information to produce enhanced 3D reconstructions of the environment [15], [17]. The downside of these techniques is that they lack the ability to tackle the non-rigid behavior of deforming objects [4], [16], [20]. Some of these techniques. e.g., for human face modeling and full-body 3D reconstruction, are restricted to very limited non-rigid behavior and require subjects to remain as rigid as possible [14], [6], [21]. To tackle these issues researchers have proposed other methods such as [27], [26] and [12], which use high quality pre-built templates or construct them as a first step and use them to track the non-rigidities and provide accurate and complete 3D recontructions.

Recently, researchers have focused on tracking highly non-rigid behaviors of deforming objects without the knowledge of any prior shape or reference [18], [7], for the purposes of, for example, depth video enhancement [9]. In our previous work, known as *KinectDeform*, we showed that a non-rigid registration method can be used in a recursive pipeline similar to *KinectFusion* to produce enhanced 3D reconstructions of deforming objects [2]. The non-rigid registration step in the pipeline is followed by surface filtering or fusion using volumetric *truncated signed distance function* (TSDF) based implicit surface representation. This surface representation scheme is view-dependent and requires organized point clouds as input. Since non-rigid registration deforms and hence destroys the organization of input point clouds, an expensive data-reorganization step in the form of meshing and ray-casting is required before surface fusion. Moreover, for fusion, a weighted average scheme is used for which parameters are chosen empirically for each iteration. Ray-casting is used again to extract the resulting point-based surface from fused *TSDF* volumes after every iteration.

In this paper, we propose a method called *View-Independent KinectDeform (VI-KinectDeform)* which improves upon the *KinectDeform* algorithm by replacing the volumetric *TSDF* based view-dependent surface representation with an octree-based view-independent and explicit surface representation using *Point Set Surfaces* based on the method of *Moving Least Squares* [3]. This results in a simplified version of *KinectDeform* with the removal of an expensive data reorganization step. Moreover, we also improve upon the fusion mechanism by proposing an automated recursive filtering scheme using a simple *Kalman filter* [10]. Due to our explicit surface representation, surface prediction step at the end of each iteration is also not required resulting in a simpler algorithm. We compare the results of *VI-KinectDeform* with those of *KinectDeform* using non-rigidly deforming objects and show that for the same number of iterations *VI-KinectDeform* produces stable and more accurate 3D reconstructions.

The remainder of this paper is organized as follows: Section 2 describes the problem at hand and gives a background on the surface representation and recursive filtering method proposed in *KinectDeform*. This is followed by an introduction to the *Point Set Surfaces* based on *MLS*. Section 3 details the proposed approach. Section 4 presents qualitative and quantitative evaluation of results of the proposed method and compares them with the results of *KinectDeform* and other methods. This is followed by a conclusion in Sect. 5.

2 Background

2.1 *Problem Formulation and KinectDeform*

At each discrete time-step $i \in \mathbb{N}$, a static or moving camera acquires a point cloud $\mathcal{V}_i$ containing a number of points $U \in \mathbb{N}$. Note that $\mathcal{V}_i$ may be organized or unorganized. The point-set $\{\mathbf{p}_j\}$ in $\mathcal{V}_i$, where $\mathbf{p}_j \in \mathbb{R}^3$ and $j \in \{1, \ldots, U\}$, approximates the underlying surface of deformable objects in camera's field of view. Considering a sequence of N such acquired point clouds $\{\mathcal{V}_0, \mathcal{V}_1, \dots, \mathcal{V}_{N-1}\}$, each acquisition $\mathcal{V}_i$ is associated with the previous acquisition $\mathcal{V}_{i-1}$ via [2]:

$$\mathcal{V}_i = h_i\left(\mathcal{V}_{i-1}\right) + \mathcal{E}_i, \tag{1}$$

where $h_i(\cdot)$ is the non-rigid deformation which deforms $\mathcal{V}_{i-1}$ to $\mathcal{V}_i$, and $\mathcal{E}_i$ represents the sensor noise and sampling errors. The problem at hand is therefore to reduce $\mathcal{E}_i$ for $i > 0$, to recover an enhanced sequence $\{\mathcal{V}_0^{f'}, \mathcal{V}_1^{f'}, \dots, \mathcal{V}_{N-1}^{f'}\}$ starting from the input sequence $\{\mathcal{V}_0, \mathcal{V}_1, \dots, \mathcal{V}_{N-1}\}$ [2]. In *KinectDeform*, we defined a recursive filtering function $f(\cdot,\cdot)$ to solve this problem which sequentially fuses the current measurement $\mathcal{V}_i$ with the result of the previous iteration $\mathcal{V}_{i-1}^{f'}$ by tracking the non-rigid deformations between them such that:

$$\mathcal{V}_i^{f'} = \begin{cases} \mathcal{V}_i & \text{for } i = 0, \\ f(\mathcal{V}_{i-1}^{f'}, \mathcal{V}_i) & i > 0. \end{cases} \tag{2}$$

As mentioned before a major shortcoming of the *KinectDeform* scheme lies in the 3D surface representation based on the view-dependent *truncated signed distance function (TSDF)* volume for data fusion and filtering [2]. Construction of a *TSDF* volume for a point cloud $\mathcal{V}_i$ requires computing a scalar *TSDF* value for each voxel represented by its centroid $\mathbf{c} \in \mathbb{R}^3$. The *TSDF* function $S_{\mathcal{V}_i}$ may be defined as follows:

$$S_{\mathcal{V}_i}(\mathbf{c}) = \Psi(\|\mathbf{c}\|_2 - \|\mathbf{p}_j\|_2), \tag{3}$$

where $j = \pi(\mathbf{Kc})$, $j \in \{1, \dots, U\}$, is projection of the centroid $\mathbf{c}$ to camera's image plane using camera's intrinsic matrix $\mathbf{K}$. This, in turn, requires the points in $\mathcal{V}_i$ to be organized with respect to the image plane, moreover:

$$\Psi(\eta) = \begin{cases} min\{1, \frac{\eta}{\mu}\} \cdot sgn(\eta) & \text{iff } \eta \geq -\mu, \\ 0 & \text{otherwise}, \end{cases} \tag{4}$$

where μ is the truncation distance and sgn is the sign function. Therefore, after non-rigid registration which destroys the data organization of our input point cloud $\mathcal{V}_i^{f'}$, an expensive data reorganization step based on meshing and ray-casting is required for computation of a *TSDF*. After that, the *TSDF* volumes created using $\mathcal{V}_i^r$ and $\mathcal{V}_i$ are fused together using an empirical weighting scheme whereby the weighting parameters are chosen manually. This is followed by another surface prediction step via ray-casting to extract the final filtered surface from the fused volume.

2.2 *Point Set Surfaces*

Keeping in view the *KinectDeform* method explained in Sect. 2.1, a simpler approach would be to replace the view-dependent *TSDF* volume-based surface representation for fusion and filtering with a view-independent surface representation. This would result in avoiding data reorganization and surface prediction steps. As mentioned before the input points $\{\mathbf{p}_j\}$ approximate the underlying surface of objects in the scene. In [3], Alexa et al. built upon Levin's work [11], and proposed a view-independent point-based surface reconstruction method based on *Moving Least Squares (MLS)*. This method projects a point $\mathbf{q}$ lying near $\{\mathbf{p}_j\}$ on the underlying surface approximated by the local neighborhood of $\mathbf{q}$. Apart from facilitating the computation of the differential geometric properties of the surface such as normals and curvatures, this method is able to handle noisy data and provides smooth reconstructions. Moreover, the local nature of projection procedure improves the efficiency of the algorithm [5].

The projection procedure as proposed by Alexa et al. is divided into two steps [3]. In the first step a local reference domain, i.e., a plane $H_{\mathbf{q}} = \{\mathbf{p} \in \mathbb{R}^3 : \mathbf{n}^T\mathbf{p} = \mathbf{n}^T\mathbf{v}\}$, $\mathbf{v}, \mathbf{n} \in \mathbb{R}^3$ and $\|\mathbf{n}\| = 1$, is computed by minimizing the following non-linear energy function [5]:

$$e_{MLS}(\mathbf{v}, \mathbf{n}) = \sum_{s_{\mathbf{q}}=1}^{U_{\mathbf{q}}} w(\|\mathbf{p}_{s_{\mathbf{q}}} - \mathbf{v}\|)\langle \mathbf{n}, \mathbf{p}_{s_{\mathbf{q}}} - \mathbf{v}\rangle^2, \tag{5}$$

where $\{\mathbf{p}_{s_{\mathbf{q}}}\} \subset \{\mathbf{p}_j\}$, $s_{\mathbf{q}} \in \{1, \ldots, U_{\mathbf{q}}\}$ and $U_{\mathbf{q}}$ is the total number of neighboring points within a fixed radius around $\mathbf{q}$. Also $\mathbf{n} = (\mathbf{q} - \mathbf{v})/\|\mathbf{q} - \mathbf{v}\|$, $\langle ., . \rangle$ is the dot product and $w(e) = \exp^{(-\frac{e^2}{d^2})}$ is the Gaussian weight function where d represents the anticipated spacing between neighboring points [3]. The surface features of size less than d are smoothed out due to the *MLS* projection. Replacing $\mathbf{v}$ by $\mathbf{q} + t\mathbf{n}$ where $t \in \mathbb{R}$ in (5) we have:

$$e_{MLS}(\mathbf{q}, \mathbf{n}) = \sum_{s_{\mathbf{q}}=1}^{U_{\mathbf{q}}} w(\|\mathbf{p}_{s_{\mathbf{q}}} - \mathbf{q} - t\mathbf{n}\|)\langle \mathbf{n}, \mathbf{p}_{s_{\mathbf{q}}} - \mathbf{q} - t\mathbf{n}\rangle^2. \tag{6}$$

The minimum of (6) is found with the smallest t and the local tangent plane $H_{\mathbf{q}}$ near $\mathbf{q}$ [3]. The local reference domain is then defined by an orthonormal coordinate system in $H_{\mathbf{q}}$ with $\mathbf{v}$ as its origin [5].

In the next step, we find the orthogonal projections of points in $\{\mathbf{p}_{s_{\mathbf{v}}}\} \subset \{\mathbf{p}_j\}$, where $s_{\mathbf{v}} \in \{1, \ldots, U_{\mathbf{v}}\}$, lying in the local neighborhood of $\mathbf{v}$ to get their corresponding 2D representations $(x_{s_{\mathbf{v}}}, y_{s_{\mathbf{v}}})$ in the local coordinate system in $H_{\mathbf{q}}$. The height of $\mathbf{p}_{s_{\mathbf{v}}}$ over $H_{\mathbf{q}}$ is found via:

$$h_{s_{\mathbf{v}}} = \langle \mathbf{n}, \mathbf{p}_{s_{\mathbf{v}}} - \mathbf{q} - t\mathbf{n}\rangle. \tag{7}$$

Using the local 2D projections and the height map, a local bivariate polynomial approximation $g : \mathbb{R}^2 \to \mathbb{R}$ is computed by minimizing the weighted least squares error:

$$\sum_{s_{\mathbf{v}}=1}^{U_{\mathbf{v}}} w(\|\mathbf{p}_{s_{\mathbf{v}}} - \mathbf{q} - t\mathbf{n}\|)(g(x_{s_{\mathbf{v}}}, y_{s_{\mathbf{v}}}) - h_{s_{\mathbf{v}}})^2. \tag{8}$$

The degree of the polynomial to be computed is fixed beforehand. At the end, projection P of $\mathbf{q}$ onto the underlying surface is defined by the polynomial value at the origin, i.e.:

$$P(\mathbf{q}) = \mathbf{v} + g(0, 0)\mathbf{n} = \mathbf{q} + (t + g(0, 0))\mathbf{n}. \tag{9}$$

The projected point is considered to be the resulting filtered point lying on the approximated surface. These two steps are repeated for all points which need to be sampled to sufficiently represent the surfaces of objects in camera's field of view to get enhanced 3D reconstructions.

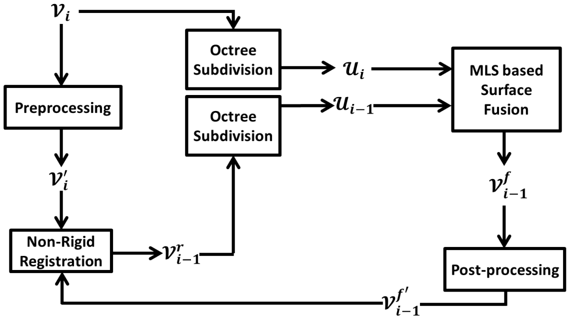


Fig. 1. Detailed pipeline of *VI-KinectDeform*. $\mathcal{V}_i$: input point cloud at time-step i. $\mathcal{V}_i^{'}$: result of pre-processing on $\mathcal{V}_i$. $\mathcal{V}_i^r$: result of non-rigid registration of $\mathcal{V}_{i-1}^{f'}$ to $\mathcal{V}_i^{'}$. $\mathcal{U}_i$ and $\mathcal{U}_{i-1}$: resulting voxel sets based on octree sub-division corresponding to $\mathcal{V}_i$ and $\mathcal{V}_{i-1}^r$ respectively. $\mathcal{V}_{i-1}^f$: the result of projection-based *MLS* surface computation and *Kalman filtering-based* fusion. $\mathcal{V}_{i-1}^f$: the final result after post-processing. For more details please read Sects. 2 and 3.

3 Proposed Technique

Figure 1 shows the pipeline of *VI-KinectDeform* which is an improved/simplified version of *KinectDeform*. After the non-rigid registration step which deforms $\mathcal{V}_{i-1}^{f'}$ to produce $\mathcal{V}_{i-1}^r$ which is mapped to $\mathcal{V}_i$, the data reorganization step is removed. Instead, a view-independent surface representation and filtering based on the *MLS* method is proposed. Since the *MLS* method works on the local neighborhoods of sampled points, voxelizing/sub-dividing the space of input 3D point clouds not only provides us with sampling information but also helps in accelerating the search for local neighborhoods of the sampled points. After that, the sampled points are projected onto the underlying surfaces of both point clouds based on the *MLS* method. The resulting projections are then fused together via an automatic *Kalman filtering* based scheme to give enhanced 3D reconstructions. These steps are explained as follows:

3.1 Sampling and *MLS* based projection

We use octree data structure to sample the space occupied by $\mathcal{V}_i$ and $\mathcal{V}_{i-1}^r$ resulting in two voxel sets $\mathcal{U}_i$ and $\mathcal{U}_{i-1}$ with a pre-defined depth $l \in \mathbb{N}$. At depth level l, $\mathcal{U}_i$ and $\mathcal{U}_{i-1}$ contain m_i^l and m_{i-1}^l non-empty voxels, respectively. It is to be noted that since $\mathcal{V}_i$ and $\mathcal{V}_{i-1}^r$ are mapped, the corresponding voxels in $\mathcal{U}_i$ and $\mathcal{U}_{i-1}$ occupy the same space. Each voxel $u_{i,a}^l \in \mathcal{U}_i$ where $a \in \{1, \dots, m_i^l\}$ (or similarly each voxel $u_{i-1,b}^l \in \mathcal{U}_{i-1}$) is represented by its geometric center $\mathbf{c}_{i,a}^l$ (or $\mathbf{c}_{i-1,b}^l$), the points contained in the voxel and information about its immediate neighbors. These centroids lying near input points provide us with suitable sampling points to be projected onto the underlying surface based on the procedure explained in Sect. 2.2. Therefore, in the next step the centroid of each non-empty leaf voxel in $\mathcal{U}_i \cup \mathcal{U}_{i-1}$ lying in the vicinity of points from both $\mathcal{V}_i$

and $\mathcal{V}^r_{i-1}$ is projected on the approximated underlying surfaces using its corresponding neighborhood points in $\mathcal{V}_i$ and $\mathcal{V}^r_{i-1}$ respectively via the *MLS* method to get:

$$\begin{aligned}\mathbf{p}_{i,k} = P_i(\mathbf{c}^l_{i,a}), \mathbf{p}_{i-1,k} = P_{i-1}(\mathbf{c}^l_{i,a}), or\\ \mathbf{p}_{i,k} = P_i(\mathbf{c}^l_{i-1,b}), \mathbf{p}_{i-1,k} = P_{i-1}(\mathbf{c}^l_{i-1,b}).\end{aligned} \tag{10}$$

The degree of the bivariate polynomial approximating the underlying surface computed for each centroid is kept variable (max. 3 for our experiments) depending on the number of points found in the neighborhood. Hence as a result of the *MLS*-based projection procedure, two sets of corresponding filtered points, $\{\mathbf{p}_{i,k}\}$ and $\{\mathbf{p}_{i-1,k}\}$, are generated.

3.2 Fusion

It is clear that under ideal conditions, i.e., sensor noise free and with perfectly registered inputs $\mathcal{V}_i$ and $\mathcal{V}^r_{i-1}$, $\{\mathbf{p}_{i,k}\}$ and $\{\mathbf{p}_{i-1,k}\}$ should be same. Therefore in this step we propose a methodology to fuse the corresponding projected points $\{\mathbf{p}_{i,k}\}$ and $\{\mathbf{p}_{i-1,k}\}$, taking into account noise factors affecting them to produce a filtered 3D reconstruction $\mathcal{V}^f_i$. In *KinectDeform* we performed a surface fusion/filtering using a weighted average of *TSDF* values of corresponding voxels [2]. The weights are chosen empirically based on an analysis of noise factors affecting the two input voxel sets per iteration. The main noise factor affecting the current measurement $\mathcal{V}_i$, and hence $\{\mathbf{p}_{i,k}\}$, is the sensor noise while on the other hand for $\mathcal{V}^r_{i-1}$ it is assumed that, due to pre-processing, some amount of this sensor noise is mitigated with some loss of details and hence the main noise factor is error due to non-rigid registration [2]. This should be coupled with iterative effects of filtering as $\mathcal{V}^r_{i-1}$ is indeed a deformed state of the filtered $\mathcal{V}^{f'}_{i-1}$.
To tackle these factors, we propose an automatic filtering approach by point tracking with a *Kalman filter* [10]. The observation model is based on the current measurements i.e. $\{\mathbf{p}_{i,k}\}$, and the associated sensor noise n^s_i is assumed to follow a Gaussian distribution $n^s_i \sim \mathcal{N}(0, \sigma^2_{s,i})$. Similarly the process/motion model is based on $\{\mathbf{p}_{i-1,k}\}$, and the associated process noise n^r_{i-1} is assumed to follow a Gaussian distribution $n^r_{i-1} \sim \mathcal{N}(0, \sigma^2_{r,i-1})$. Therefore the prediction step is:

$$\begin{cases}\mathbf{p}_{i|i-1,k} = \mathbf{p}_{i-1,k},\\ \sigma^2_{i|i-1} = \sigma^2_{i-1|i-1} + \sigma^2_{r,i-1},\end{cases} \tag{11}$$

and measurement update is given as:

$$\begin{cases}\mathbf{p}_{i|i,k} = \mathbf{p}_{i|i-1,k} + k_i(\mathbf{p}_{i,k} - \mathbf{p}_{i|i-1,k}),\\ \sigma^2_{i|i} = \sigma^2_{i|i-1} - k_i\sigma^2_{i|i-1},\end{cases} \tag{12}$$

where:

$$k_i = \frac{\sigma^2_{i|i-1}}{\sigma^2_{i|i-1} + \sigma^2_{s,i}}. \tag{13}$$

This results in the filtered set of points $\{\mathbf{p}_{i|i,k}\}$ which constitutes $\mathcal{V}^f_i$.

4 Experiments and Results

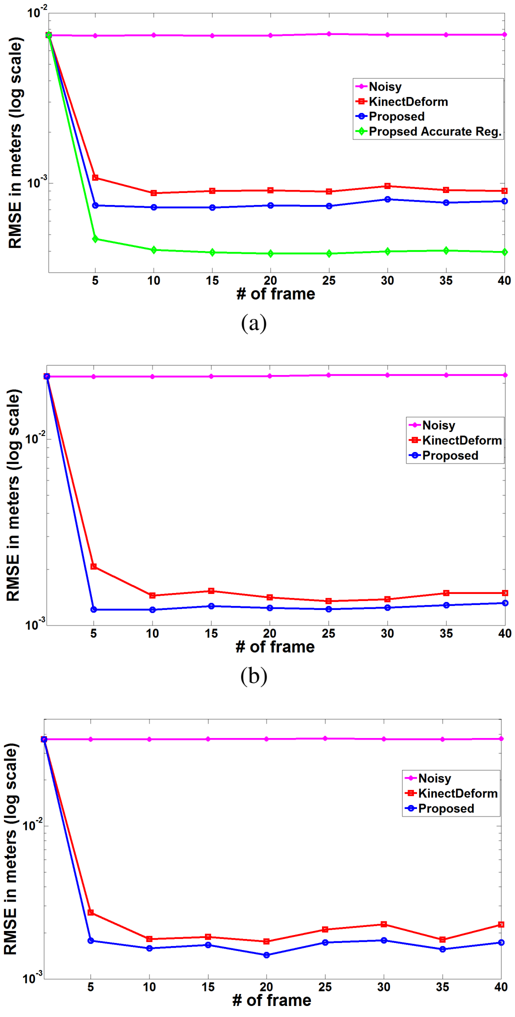
The quality of *VI-KinectDeform* is analyzed both quantitatively and qualitatively. We use the "Facecap" dataset which captures a person's face deforming non-rigidly due to changing expressions in different scenes [23]. The selected scene includes 40 frames. We simulate a depth camera in *V-Rep* [1], placed approximately at $0.5\ m$ away from the object and add *Gaussian* noise with zero mean and standard deviations of $0.01\ m$, $0.03\ m$ and $0.05\ m$, respectively. Experiments are carried out using these datasets for both *VI-KinectDeform* and *KinectDeform*. A bilateral filter is used in the pre-processing step to obtain improved registration for both methods [22]. We use the algorithm proposed by Destelle et al. [7] for non-rigid registration in both methods. We use the proposed automated fusion scheme in both *VI-KinectDeform* and *KinectDeform* by replacing the empirical fusion scheme used previously. Post-processing is based on the bilateral mesh de-noising with very small parameters for the neighborhood size and the projection distance for both *VI-KinectDeform* and *KinectDeform* [8].

The quantitative evaluation of *VI-KinectDeform* as compared to *KinectDeform* is reported in Fig. 2. It shows the root mean square error (RMSE) of the data enhanced with *VI-KinectDeform*, and the data enhanced with *KinectDeform* with respect to the ground truth data for different noise levels. These results show superior performance of *VI-KinectDeform* in terms of overall accuracy of 3D reconstructions as compared to *KinectDeform*. It is noted that the accuracy of the proposed technique is restricted by the accuracy of the considered non-rigid registration algorithm. We have tested our proposed *VI-KinectDeform* by using non-rigid registration parameters obtained from noise free data. Post-processing step is skipped in this case. The resulting curve in Fig. 2(a) shows a significant decrease in error when using *VI-KinectDeform* as compared to its earlier version. This is observed through all frames. The qualitative analysis presented in Fig. 3, corresponding to the noise level and results in Fig. 2(a), shows superior quality of 3D reconstructions obtained via *VI-KinectDeform* in terms of feature preservation and smoothness when compared to the results obtained via *KinectDeform*.

For further analysis of performance of the proposed technique, we use the "Swing" dataset [24]. We, again, simulate a depth camera in *V-Rep* placed approximately at $1.5\ m$ away from the object and add *Gaussian* noise with zero mean and standard deviation of $0.0075\ m$. We use 20 frames for this experiment. We analyze the performance of the proposed *VI-KinectDeform* with 3 other view-independent surface representation schemes. These representation schemes are based on finding the surface approximation with respect to each centroid belonging to the leaf nodes of $\mathcal{U}_i$ and $\mathcal{U}_{i-1}$ lying close to $\mathcal{V}_i$ and $\mathcal{V}^r_{i-1}$.

The first scheme is based on finding the closest points in local neighborhoods of the centroids. The second scheme is based on finding the weighted mean of all points lying in local neighborhoods of each centroid using the weighting scheme similar to the one used in (5). The third scheme fits tangent planes to points in local neighborhoods and finds the projections of the centroid on them. It is similar to the proposed scheme wherein the degree of the polynomial is fixed to one.

Quantitative and qualitative results are shown in Fig. 4 and Fig. 5, respectively. As expected, Fig. 4 shows that the closest point-based method is least accurate followed by the weighted mean-based method, the plane projection-based method, and the proposed



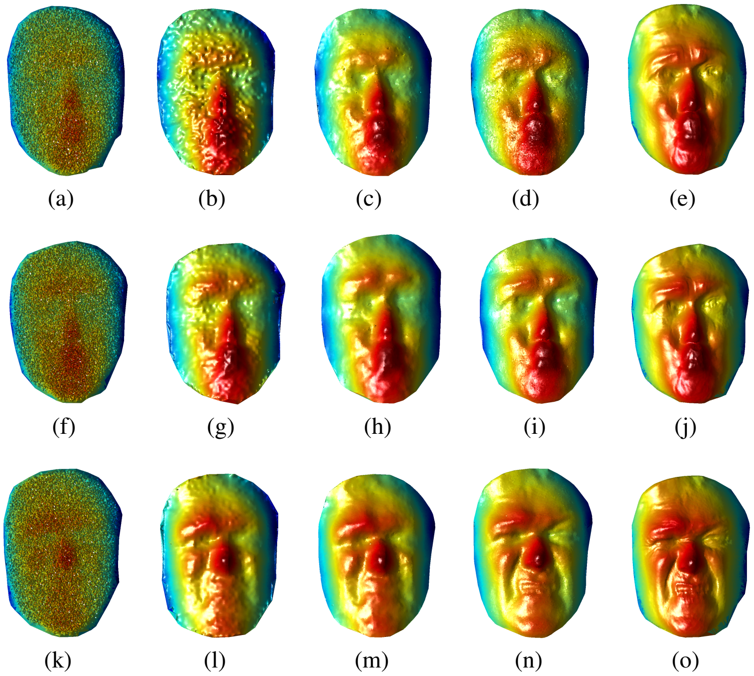
(a)

(b)

(c)

Fig. 2. "Facecap" dataset. Quantitative analysis on data with different levels of *Gaussian* noise. Each figure contains RMSE in log scale of: noisy data, result of *KinectDeform* and result of *VI-KinectDeform*. (a) Results for *Gaussian* noise with standard deviation of $0.01\ m$. It also contains RMSE in log scale of *VI-KinectDeform* with registration based on noise free data. (b) Results for *Gaussian* noise with standard deviation of $0.03\ m$. (c) Results for *Gaussian* noise with standard deviation of $0.05\ m$.

projection-based *MLS* method in terms of overall accuracy. Similar results are obtained via quantitative analysis as shown in Fig. 5 wherein the proposed method produces the most accurate and feature preserving reconstruction. Plane projection-based method also gives good results but small features such as nose and curves on clothing are not well preserved. This experiment also shows that the proposed pipeline is generic enough



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

(k) (l) (m) (n) (o)

Fig. 3. "Facecap" dataset. **First row:** Frame #5, **Second row:** Frame #15, **Third row:** Frame #35. Each row contains noisy data with *Gaussian* noise of standard deviation $0.01\ m$, result of *KinectDeform*, result of *VI-KinectDeform*, result of *VI-KinectDeform* with registration based on noise free data and ground truth respectively.

such that any view-independent point-based surface representation scheme using local neighborhoods can replace the proposed *MLS*-based scheme.

5 Conclusion and Future Work

In this work we have proposed *VI-KinectDeform*, an automated recursive filtering scheme for producing enhanced 3D reconstructions of non-rigidly deforming objects. It improves upon our previous work, i.e., *KinectDeform* [2], by replacing the implicit view-dependent *TSDF* based surface representation scheme with an explicit *MLS*-based view-independent surface representation scheme [3]. This simplifies the pipeline by removing surface prediction and extraction steps. Moreover we improve upon the data fusion scheme by proposing an automated point tracking with a *Kalman filter* [10], The quantitative and qualitative evaluation of our method shows that it is able to produce smooth and feature preserving 3D reconstructions with an improved accuracy when compared to *KinectDeform*. We also show that the proposed pipeline is generic, and can use any view-independent point-based surface representation scheme. The generic and view-independent nature of this algorithm allows for the extension to a multi-view system

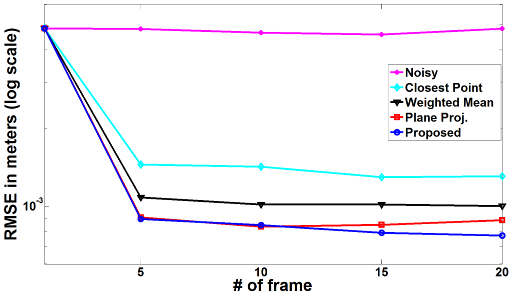


Fig. 4. “Swing” dataset. RMSE in log scale of: noisy data with *Gaussian* noise of standard deviation 0.0075 m, result of closest point-based surface representation, result of weighted-mean based surface representation, result of local plane projection-based surface representation and result of the proposed projection-based *MLS* surface representation. Please read Sect. 4 for more details.

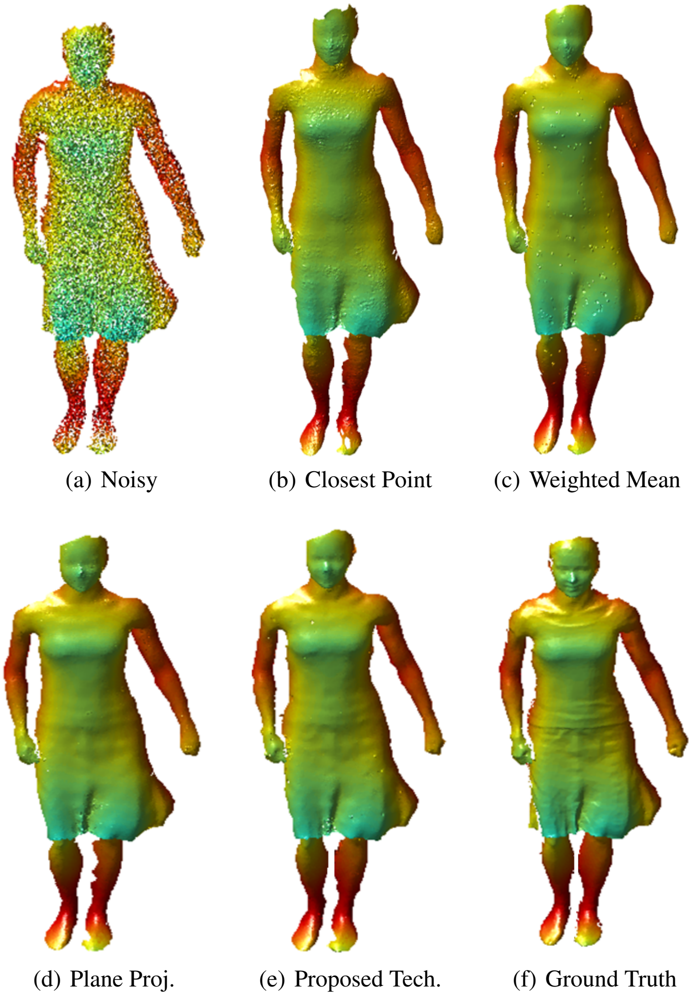
to produce complete 360° enhanced 3D reconstructions of scenes containing non-rigid objects. This constitutes our future work.

Acknowledgment

The “Facecap” dataset was provided courtesy of the *Graphics, Vision & Video Research Group of the Max Planck Institute for Informatics* [23]. This work was supported by the National Research Fund (FNR), Luxembourg, under the CORE project C11/BM/1204105/FAVE/Ottersten.

References

1. V-REP, http://www.coppeliarobotics.com/
2. Afzal, H., Ismaeil, K.A., Aouada, D., Destelle, F., Mirbach, B., Ottersten, B.: KinectDeform: Enhanced 3D Reconstruction of Non-Rigidly Deforming Objects. In: 3DV Workshop on Dynamic Shape Measurement and Analysis. Tokyo, Japan (2014)
3. Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, C.T.: Computing and rendering point set surfaces. Visualization and Computer Graphics, IEEE Transactions on 9(1), 3–15 (Jan 2003)
4. Bylow, E., Sturm, J., Kerl, C., Kahl, F., Cremers, D.: Real-Time Camera Tracking and 3D Reconstruction Using Signed Distance Functions. In: Robotics: Science and Systems Conference (RSS) (June 2013)
5. Cheng, Z.Q., Wang, Y.Z., Li, B., Xu, K., Dang, G., Jin, S.Y.: A survey of methods for moving least squares surfaces. In: Proceedings of the Fifth Eurographics / IEEE VGTC Conference on Point-Based Graphics. pp. 9–23. SPBG’08, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland (2008)
6. Cui, Y., Chang, W., Nll, T., Stricker, D.: KinectAvatar: Fully Automatic Body Capture Using a single Kinect. In: ACCV Workshop on Color Depth fusion in computer. ACCV (2012)
7. Destelle, F., Roudet, C., Neveu, M., Dipanda, A.: Towards a real-time tracking of dense point-sampled geometry. International Conference on Image Processing pp. 381–384 (2012)



(a) Noisy (b) Closest Point (c) Weighted Mean

(d) Plane Proj. (e) Proposed Tech. (f) Ground Truth

Fig. 5. “Swing” dataset. **First row:** *Left:* noisy data with *Gaussian* noise of standard deviation $0.0075\ m$, *Center:* result of closest point-based surface representation, *Right:* result of weighted mean-based surface representation. **Second row:** *Left:* result of local plane projection-based surface representation, *Center:* result of the proposed projection-based *MLS* surface representation, *Right:* ground truth.

8. Fleishman, S., Drori, I., Cohen-Or, D.: Bilateral mesh denoising. In: ACM SIGGRAPH 2003 Papers. pp. 950–953. SIGGRAPH ’03, ACM, New York, NY, USA (2003)
9. Ismaeil, K.A., Aouada, D., Solignac, T., Mirbach, B., Ottersten, B.: Real-Time Non-Rigid Multi-Frame Depth Video Super-Resolution. In: CVPR Workshop on Multi-Sensor Fusion for Dynamic Scene Understanding. Boston, MA, USA (2015)
10. Kalman, R.E.: A new approach to linear filtering and prediction problems. Transactions of the ASME–Journal of Basic Engineering 82(Series D), 35–45 (1960)

11. Levin, D.: Mesh-independent surface interpolation. In: Brunnett, H., Mueller (eds.) Geometric Modeling for Scientific Visualization. pp. 37–49. Springer-Verlag (2003)
12. Li, H., Adams, B., Guibas, L.J., Pauly, M.: Robust Single-view Geometry and Motion Reconstruction. In: ACM SIGGRAPH Asia 2009 Papers. pp. 175:1–175:10. SIGGRAPH Asia '09, ACM, New York, NY, USA (2009), http://doi.acm.org/10.1145/1661412.1618521
13. Mac Aodha, O., Campbell, N.D., Nair, A., Brostow, G.J.: Patch Based Synthesis for Single Depth Image Super-Resolution. In: ECCV (3). pp. 71–84 (2012)
14. Mrcio, C., Apolinario Jr., A.L., Souza, A.C.S.: KinectFusion for Faces: Real-Time 3D Face Tracking and Modeling Using a Kinect Camera for a Markerless AR System. SBC Journal on 3D Interactive Systems 4, 2–7 (2013)
15. Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohli, P., Shotton, J., Hodges, S., Fitzgibbon, A.: KinectFusion: Real-Time Dense Surface Mapping and Tracking. In: Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality. pp. 127–136. ISMAR '11, IEEE Computer Society, Washington, DC, USA (2011), http://dx.doi.org/10.1109/ISMAR.2011.6092378
16. Nießner, M., Zollhöfer, M., Izadi, S., Stamminger, M.: Real-time 3D Reconstruction at Scale using Voxel Hashing. ACM Transactions on Graphics (TOG) (2013)
17. Roth, H., Vona, M.: Moving Volume KinectFusion. In: Proceedings of the British Machine Vision Conference. pp. 112.1–112.11. BMVA Press (2012)
18. Rouhani, M., Sappa, A.: Non-rigid shape registration: A single linear least squares framework. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) Computer Vision ECCV 2012, Lecture Notes in Computer Science, vol. 7578, pp. 264–277. Springer Berlin Heidelberg (2012)
19. Schuon, S., Theobalt, C., Davis, J., Thrun, S.: LidarBoost: Depth Superresolution for ToF 3D Shape Scanning. In Proc. of IEEE CVPR 2009 (2009)
20. Steinbrcker, F., Kerl, C., Cremers, D.: Large-Scale Multi-resolution Surface Reconstruction from RGB-D Sequences. In: Proceedings of the 2013 IEEE International Conference on Computer Vision. pp. 3264–3271. ICCV '13, IEEE Computer Society, Washington, DC, USA (2013), http://dx.doi.org/10.1109/ICCV.2013.405
21. Sturm, J., Bylow, E., Kahl, F., Cremers, D.: CopyMe3D: Scanning and printing persons in 3D. In: German Conference on Pattern Recognition (GCPR). Saarbrücken, Germany (September 2013)
22. Tomasi, C., Manduchi, R.: Bilateral Filtering for Gray and Color Images. In: Proceedings of the Sixth International Conference on Computer Vision. pp. 839–. ICCV '98, IEEE Computer Society, Washington, DC, USA (1998)
23. Valgaerts, L., Wu, C., Bruhn, A., Seidel, H.P., Theobalt, C.: Lightweight Binocular Facial Performance Capture Under Uncontrolled Lighting. ACM Trans. Graph.
24. Vlasic, D., Baran, I., Matusik, W., Popović, J.: Articulated mesh animation from multi-view silhouettes. In: ACM SIGGRAPH 2008 Papers. pp. 97:1–97:9. SIGGRAPH '08, ACM, New York, NY, USA (2008)
25. Wang, K., Wang, X., Pan, Z., Liu, K.: A two-stage framework for 3d facereconstruction from rgbd images. Pattern Analysis and Machine Intelligence, IEEE Transactions on 36(8), 1493–1504 (Aug 2014)
26. Zeng, M., Zheng, J., Cheng, X., Jiang, B., Liu, X.: Dynamic human surface reconstruction using a single kinect. In: Computer-Aided Design and Computer Graphics (CAD/Graphics), 2013 International Conference on. pp. 188–195 (Nov 2013)
27. Zollhöfer, M., Nießner, M., Izadi, S., Rehmann, C., Zach, C., Fisher, M., Wu, C., Fitzgibbon, A., Loop, C., Theobalt, C., Stamminger, M.: Real-time Non-rigid Reconstruction using an RGB-D Camera. ACM Transactions on Graphics (TOG) (2014)