Gradient Flow Evolution for 3D Fusion From a Single Depth Sensor

Jiwoo Kang, Seongmin Lee, Mingyu Jang, and Sanghoon Lee, *Senior Member, IEEE*

***Abstract*— We present a novel real-time framework for non-rigid 3D reconstruction that is robust to noise, camera poses, and large deformation from a single depth camera. KinectFusion has achieved high-quality 3D object reconstructions in real-time by implicitly representing an object's surface with a signed distance field (SDF) representation from a single depth camera. Many studies for incremental reconstruction have been presented since then, with the surface estimation improving over time. Previous works primarily focused on improving conventional SDF matching and deformation schemes. In contrast to these works, the proposed framework tackles the problem of temporal inconsistency caused by SDF approximation and fusion to manipulate SDFs and reconstruct a target more accurately over time. In our reconstruction pipeline, we introduce a refinement evolution method, where an erroneous SDF from a depth sensor is recovered more accurately in a few iterations by propagating erroneous SDF values from the surface. Reliable gradients of refined SDFs enable more accurate non-rigid tracking of a target object. Furthermore, we propose a level-set evolution for SDF fusion, enabling SDFs to be manipulated stably in the reconstruction pipeline over time. The proposed methods are fully parallelizable and can be executed in real-time. Qualitative and quantitative evaluations show that incorporating the refinement and fusion methods into the reconstruction pipeline improves 3D reconstruction accuracy and temporal reliability by avoiding cumulative errors over time. Evaluation results show that our pipeline results in more accurate reconstruction that is robust to noise and large motions, as well as outperforms previous state-of-the-art reconstruction methods.**

***Index Terms*— 3D reconstruction, signed distance field, implicit representation, non-rigid deformation, incremental reconstruction.**

I. Introduction

With the advancement of structured-light and time-of-flight sensors and the broader availability of affordable sensors [4] such as the Microsoft Kinect [5] and Intel ReslSense [6], techniques for reconstructing a three-dimensional (3D) surface model from a sequence of depths have been widely studied [7], [8]. Depth measurements obtained from these commercial sensors are noisy and incomplete and contain numerous outliers.

KinectFusion [1], [2] has made a breakthrough in which the target surface estimation is improved incrementally over time from a single depth camera, achieving high-quality 3D reconstructions in real-time. In KinectFusion, a truncated signed distance field (TSDF) is updated incrementally by predicting the camera poses using iterative closest points (ICP) [9], [10] from a given frame. Since then, various techniques for capturing static environments have yielded impressive results [11]–[15] using TSDFs. DynamicFusion [16] is the first work that performed a non-rigid 3D reconstruction in real-time from a single depth camera by optimizing a coarse-scale warping field on a TSDF. Many works have improved DynamicFusion using more features such as color features [17], [18], albedo [19], and template models [20], [21]. Subsequent works [3], [22] have proposed variational methods that accurately track a voxel-wise warp field and handle topological changes. Recently, a 3D reconstruction pipeline has been proposed by merging and swapping 3D clusters using segmentation to better reconstruct more dynamic scenes [23]. A volumetric structure manipulation method has been introduced to handle topological changes more efficiently [24]. Some other works have used priors from a human body model for volumetric capture of a human [25], [26]. They have tried to non-rigidly register TSDFs constructed temporally from a moving object to improve 3D surface estimation.

TSDFs used in those works have a significant benefit in manipulating 3D objects directly on voxel grids by representing 3D surfaces implicitly. TSDF values are approximated by the distance to the object surfaces on the basis that the projected value into the depth map of a 3D point is equal to its projective depth if the 3D point is on a 3D surface. However, this approximation does not match when the point is far from the surface, resulting in significant approximation errors. Deformation optimization using discontinuous and inaccurate gradients leads to the misalignment of TSDFs and, thus, erroneous TSDFs with artifacts. Errors from the deformation of TSDFs are "incrementally" accumulated over time in an integration procedure of TSDFs [27], denoted as "fusion," cumulatively decreasing the performance of the reconstruction pipeline in consecutive frames. As a result, in contrast to previous works that primarily focused on TSDF registration

Manuscript received April 5, 2021; revised May 21, 2021; accepted June 5, 2021. Date of publication June 16, 2021; date of current version April 5, 2022. This work was supported in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MSIT) under Grant 2020R1A2C3011697 and in part by the Yonsei University Research Fund of 2021 under Grant 2021-22-0001. This article was recommended by Associate Editor W. Li. *(Corresponding author: Sanghoon Lee.)*

Jiwoo Kang is with the Department of IT Engineering, Sookmyung Women's University, Seoul 04310, South Korea (e-mail: jwkang@sookmyung.ac.kr).

Seongmin Lee and Mingyu Jang are with the Department of Electrical and Electronic Engineering, Yonsei University, Seoul 03722, South Korea (e-mail: lseong721@yonsei.ac.kr; jmg1002@yonsei.ac.kr).

Sanghoon Lee is with the Department of Electrical and Electronic Engineering and the Department of Radiology, Yonsei University, Seoul 03722, South Korea, and also with the College of Medicine, Yonsei University, Seoul 03722, South Korea (e-mail: slee@yonsei.ac.kr).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2021.3089695.

Digital Object Identifier 10.1109/TCSVT.2021.3089695

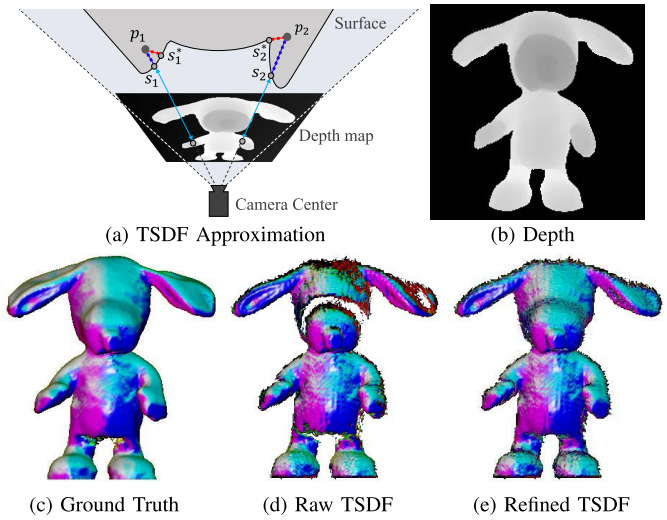


Fig. 1. (a) The procedure of TSDF approximation [1], [2], and (b)−(e) an example of the proposed refinement method on "Snoopy" in the deformable 3D reconstruction dataset [3]. The proposed evolution method efficiently refines erroneous TSDFs (d) approximated from a depth map (b) in a few iterations closer to the original surface as shown in (e). The ground truth (c) is the full 3D shape of the target, which is constructed using multiview sequences and visualized for comparison.

and deformation, we focus on the noise and incorrectness that TSDFs inherently have as they are generated from a depth map, which can lead to inaccurate TSDF registration. In addition, we address the problem of TSDF fusion using level-set evolution to reliably integrate TSDFs and preserve the signed distance property of the fused TSDF over time. Thus, in this paper, we propose a novel pipeline for non-rigid 3D reconstruction captured from a single depth camera in real-time, introducing two meaningful steps in the pipeline to recover an accurate SDF from a raw SDF and to make TSDFs reliable over time.

Fig. 1 (a) describes an approximation procedure of the signed distance from a depth camera. When a depth map is captured using a sensor, signed distances of 3D points in voxel grids from an object's surface are approximated. The signed distance of a point is trivially approximated by measuring the difference between its projective depth (z-coordinate value of the point) and a value obtained by projecting on the depth map [1], [2]. For example, if the depth map was ideally measured without noise, the projective depth of the surface point s_1 in Fig. 1 (a) and its sampling value of the depth map on the projection point have the same value. The distance of the point p_1 from the surface ($\overline{s_1^* p_1}$) is approximated using the length of the line $\overline{s_1 p_1}$ (blue dotted line), which is calculated using the difference between the projective depth of p_1 and its depth projection value (i.e., the projective depth of the surface point s_1).

However, since this approximation assumes that 3D points are near the surface and the projection rays are almost parallel with the surface normal, the procedure inherently produces erroneous TSDFs in practice. An example of an erroneous distance approximation (point p_2) from the surface is shown in Fig. 1 (a), where the approximated distance ($\overline{s_2 p_2}$) is far from the real distance ($\overline{s_2^* p_2}$) from the surface.

To tackle the underlying 3D fusion problem using TSDFs, a novel evolution method for TSDF refinement is proposed for our pipeline. The proposed refinement method is motivated by the property of the TSDF approximation procedure. [1], [2]. In other words, the point closest to an object's surface has a more reliable TSDF value. Thus, TSDF values are iteratively propagated from the direction of the object surface in the refinement evolution. The evolution method efficiently refines a raw TSDF, as shown in Fig. 1 (d), which is approximated from a depth map in Fig. 1 (b), and produces the refined TSDF in Fig. 1 (e) within a few iterations. For efficient comparison, the reference target mesh constructed using multiview sequences is also visualized in Fig. 1 (c). It is demonstrated in this paper that more accurate and reliable measurements from the refined TSDF offer significant advantages in the performance of TSDF deformation, resulting in a stable and accurate 3D reconstruction pipeline over time.

Furthermore, we propose a level-set evolution method for a 3D fusion procedure to address the problem of the TSDF fusion scheme [27], which is conventionally used for incremental 3D reconstruction. We address the temporal stability of 3D incremental reconstruction using a more reliable TSDF fusion pipeline. Rather than directly using the linear fusion of the TSDFs in time, the TSDF values are propagated along the gradient flow with their iso-distances regularized in the proposed fusion evolution of our pipeline, enabling the implicit surface property to be efficiently preserved and the object surface to be accurately represented on a TSDF in time. Fig. 2 shows an example of the proposed fusion method, where the evolution provides greater reliability to the reconstruction pipeline while accurately preserving details when compared with the conventional fusion scheme. Although each of the refinement and fusion evolutions is beneficial, it has been demonstrated that incorporating both evolutions is most beneficial to the reconstruction pipeline because their outputs are highly related in the pipeline. The proposed pipeline efficiently manipulates a TSDF so that the TSDF error does not accumulate over time. It enables accurate TSDF deformation with a few energy definitions, enabling the proposed framework to outperform state-of-the-art methods.

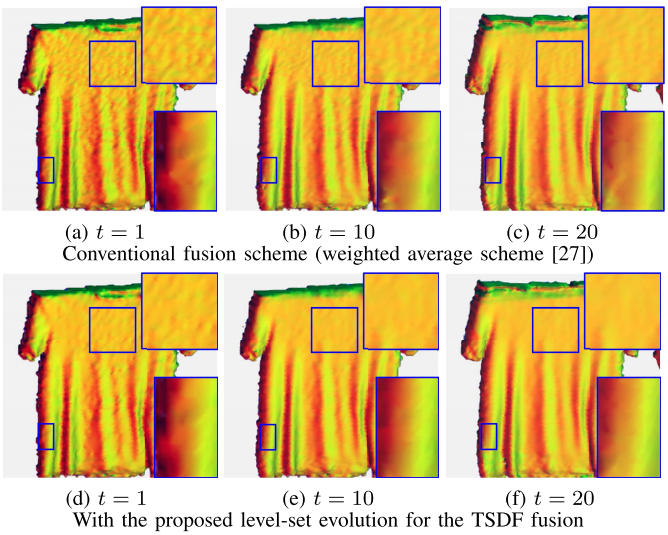
In summary, we propose a novel real-time pipeline for non-rigid 3D reconstruction that is robust to the noise, camera pose, and large deformation by temporally preserving the reliable implicit surface, in which

- A TSDF refinement method recovers TSDF values in a few iterations by propagating the TSDF values from the surface direction, resulting in the accurate deformation,
- A TSDF propagation method for TSDF fusion evolves the gradient flow of TSDFs with distance regularization, enabling the reconstruction framework to be temporally and reliably manipulated, and
- Our pipeline's procedures are variational and fully parallelizable in real-time, making them simple to implement.

II. Related Work

A. Real-Time Static Reconstruction

3D representations such as 3D points [28], [29], voxels [15], and meshes [30] have been used to reconstruct 3D



(a) $t = 1$ (b) $t = 10$ (c) $t = 20$
Conventional fusion scheme (weighted average scheme [27])

(d) $t = 1$ (e) $t = 10$ (f) $t = 20$
With the proposed level-set evolution for the TSDF fusion

Fig. 2. An example of the proposed evolution method on "Shirt" in the VolumeDeform dataset [18]. The fusion scheme conventionally used for incremental reconstruction does not sufficiently consider the temporal stability of TSDF representation. With the proposed fusion evolution, unstable values from noisy depths and alignment errors are efficiently excluded in time, whereas high-frequency components such as wrinkles are accurately preserved.

shapes from a stream of depth maps offline [31], [32]. In these methods, the correspondences between 3D objects are found by searching for the closest points [9], [10] from each source in every iteration. These methods require expensive computational costs to be realized in real-time. In addition, they are negatively affected by noise and extensive motions. Many studies [27], [33]–[35] have shown that implicit surface representations are superior in efficiently handling noise and large scene movements.

With the benefits of 3D surface representation, real-time 3D reconstruction methods [36], [37] have been introduced, including KinectFusion [1], [2] in which the dense 3D estimation is improved incrementally over time based on TSDFs [27]. In several subsequent studies, KinectFusion has been extended to large-scale scenes [13]–[15] and improved for more accurate reconstruction [11], [12].

Nevertheless, those methods did not consider underlying errors caused by TSDF generation from a depth map. They rather instead attempted find better matches for the given TSDFs. The distance from the surface along the projected ray to the depth map is used to approximate TSDFs. Thus, the TSDF values are affected considerably depending on the viewing direction and the surface normal. Consequently, we introduce a TSDF propagation method from the normal flow that is robust to noise and viewing direction to generate TSDFs.

B. Non-Rigid Incremental Reconstruction

In incremental reconstruction methods, non-rigid surface deformations are tracked temporally, and surface estimation is improved over time. In many studies on non-rigid incremental reconstruction [38]–[42], calibrated multi-cameras have been used as input sensors. On the other hand, several other studies [43]–[50] have used 3D template models for coarse estimation before reconstructing the target on a fine-scale. However, those methods require a strict camera calibration process and they are far from online applications.

Newcombe *et al.* [16] introduced DynamicFusion, the first non-rigid method for online surface reconstruction using a single Kinect depth sensor. DynamicFusion finds a warping field of non-rigid motion using an implicit surface representation. Subsequent studies [17]–[19] have improved tracking performances over DynamicFusion using image features from the corresponding images such as SIFT and surface albedo. However, in practice, the color correspondence to the depth map using off-the-shelf RGB-D sensors is inaccurate due to noise and coarse sensor resolution. Subsequent works to more accurately reconstruct dynamic scene from a single sensor have been conducted using a segmentation method [23], a volumetric structure to efficiently handle topological changes [24], human body priors [25], [26], and variational formulations for voxel-wise warping [3], [22].

Similar to the works of Slavcheva *et al.* [3], [22], our framework estimates a dense deformation field via variational minimization. Rather than employing various features or regularizers, we improve the tracking of the dense field with a few deformation energy definitions, and this is accomplished by tackling inaccuracies inherent in the generation and fusion of TSDFs. Thus, the proposed TSDF evolution for the generation and fusion can be easily combined with other reconstruction methods that use TSDFs.

C. Variational Level Set

As level set methods [51]–[54] inherently manipulate topology changes and cope skillfully with larger deformations, they have been used in various applications such as image segmentation, registration [55]–[58], and 3D surface representation [59]–[62]. Level-set based methods [3], [22] have been proposed to evolve a scene flow rather than evolving the implicit function directly for incremental reconstruction.

It is important for those variational methods that use an implicit function to preserve a signed distance from the surface during optimization. In other words, the gradient magnitude of the implicit function should to be kept on a unit scale to achieve stable results [53], [54]. As iterations for level-set evolution significantly changes the gradient in practice, many regularization and re-initialization methods [51], [63]–[67] to preserve the gradient magnitude have been proposed for the level-set methods.

Reconstruction methods [16]–[18], [38], [39] using TSDFs inherit the problem of implicit representation. Our framework addresses the requirement of preserving the implicit representation property to create a temporally reliable reconstruction pipeline. The raw TSDF approximated from a depth sensor has lots of incomplete and inaccurate values and gradients. Owing to inaccurate values, a target object cannot be represented using TSDF representation accurately to track and deform the object with temporal stability. TSDF errors are temporally accumulated by the fusion procedure from erroneous TSDFs during fusion procedures, eventually resulting in pipeline

divergence. In the following, we introduce a novel reconstruction pipeline to temporally preserve the TSDF property by addressing the limitations of TSDF construction and fusion schemes that have been conventionally used in incremental reconstruction frameworks. The two evolution methods proposed for TSDF refinement and fusion enable temporally consecutive fusions to be more robust and reliable, and thus, accurate 3D reconstruction from a single depth sensor.

III. Reconstruction Framework

A. Overview

Similar to other non-rigid 3D reconstruction methods using TSDFs [3], [16]–[18], [22], [38], [39], our reconstruction framework from a single depth stream comprises three steps: a TSDF *generation* step from a new frame, a *deformation* step to the canonical coordinates, and a *fusion* step of deformed TSDFs to the reference TSDF in the canonical coordinates. The overview pipeline of the proposed framework is illustrated in Fig. 3. After obtaining a motion field Ψ_t of the current TSDF ϕ_t that warps the canonical TSDF through the *defomation* step, a user is provided with a live frame visualization ϕ'^{fuse}_{t+1} by deforming the fused TSDF ϕ^{fuse}_{t+1} backward the motion field.

A depth map from a sensor is noisy and incomplete, and TSDF values approximated far from an object's surface are erroneous. This yields several mismatch voxels among TSDFs in the *deformation* step. These mismatches among TSDFs used in the following time iterations accumulate TSDF errors, worsening the overall reliability and accuracy of the reconstruction pipeline over time.

In the proposed framework, two novel evolutions *refinement* and *fusion* are introduced to efficiently manipulate and yield reliable TSDF values over time. The values of a raw TSDF ϕ_t are evolved by propagating the TSDF values from the normal flow before the *deformation* step in our pipeline. The TSDF evolution from the normal flow refines the raw TSDF to be more accurate and complete and TSDF voxels to have unit gradients and view-invariant values, i.e., prevents bias in captured view. Thus, the refined TSDF ϕ^{refine}_t can be registered more accurately to the reference TSDF using simple energy definition for data and smooth terms in the *deformation* step.

However, TSDFs may not be perfectly aligned through deformation, especially when capturing a largely dynamic scene. The subsequent *fusion* procedure between the deformed and canonical TSDFs can cause artifacts in the fused TSDF. The artifacts in the reference frame cause more significant misalignment and artifacts in consequent frames, eventually resulting in tracking failure.

We address the temporally reliable fusion procedure *fusion* limitation using the gradient propagation scheme of the canonical TSDF toward the fused TSDF. The level-set *fusion* through the propagation prevents significant topology changes caused by the misalignment of TSDFs while preserving the signed distance property of the TSDFs to be regularized.

In the following, we first introduce mathematical symbols, definitions, and notations by over-viewing TSDF generation from a depth stream (Sec. III-B). Subsequently, TSDF propagation for refinement by evolving TSDFs from the normal flow is presented (Sec. III-C). Then, the deformation energies for tracking TSDFs are presented (Sec. III-D) Consequently, the level-set evolution for TSDF fusion using gradient propagation is proposed (Sec. III-E).

B. Preliminaries

A signed distance [54] from an arbitrary point to an object surface can be approximated using a difference from its projection to a depth map, assuming a depth sensor is calibrated. Because this approximation is valid only near the object surface, the signed distance over the predefined threshold is truncated [68]. For discretized cubic voxels $\mathbf{x} = (x, y, z) \in \mathbb{N}^3$ corresponding to 3D points in coordinates $\mathbf{X} = (X, Y, Z) \in \mathbb{R}^3$, TSDF $\phi(\mathbf{x})$ is constructed by measuring the signed distances. The signed distance $\text{dist}(\mathbf{x})$ is approximated by projecting the points into the depth map D [1], [2] as

$$\text{dist}(\mathbf{x}) = D\left(\Pi(\mathbf{x})\right) - Z, \tag{1}$$

$$\phi(\mathbf{x}) = sgn\left(\text{dist}(\mathbf{x})\right) \cdot \min\left(\left|\text{dist}(\mathbf{x})\right|, \tau\right) \big/ \tau, \tag{2}$$

$$\omega(\mathbf{x}) = \begin{cases} 1.0, & \text{if } \text{dist}(\mathbf{x}) > -\tau \\ 0.0, & \text{otherwise} \end{cases} \tag{3}$$

where $\Pi : \mathbb{R}^3 \rightarrow \mathbb{N}^2$ is the projection operator to pixels on the depth map from 3D coordinates, $sgn(\cdot)$ is the sign operator, τ is the truncated margin, and $\omega(\mathbf{x})$ is the TSDF weight. The TSDF representation is easily converted to mesh representation through the marching cube algorithm [69].

When TSDFs constructed from consecutive depth frames are given, they can be fused using the weighted average scheme [27]. The truncated margin τ guarantees TSDF accuracy by ignoring regions far from the object surface and determines the expected thickness of the object surface. It is necessary to set the truncated margin sufficiently to represent TSDFs accurately near the surface and evolve the gradient fields. However, an extremely large margin makes the boundaries between the TSDFs to be smooth in the fusion procedure. The truncated margin is determined depending on the resolutions of a TSDF volume, depth map, and depth sensor noise in practice. In our experiments, the margin τ is set to five times v_s, where v_s is the voxel size, an actual distance between grid voxels.

It is assumed that a single depth sensor is used and both the target shape and the camera position change over time. Our system reconstructs the target surface, while improving surface estimation over time for the non-rigid target. For a depth map in time t (D_t), the TSDF $\phi_t(\mathbf{x})$ and the corresponding weight $\omega_t(\mathbf{x})$ volumes are constructed using (2) and (3). However, the raw TSDF ϕ_t is inherently incomplete and noisy. For reliable reconstruction, the raw TSDF is refined to accurately represent the signed distance from the surface through the TSDF evolution from the normal flow before TSDF deformation.

C. TSDF Evolution for Refinement

The core motivation of the proposed method is from the fact that only the approximated values of TSDFs close to

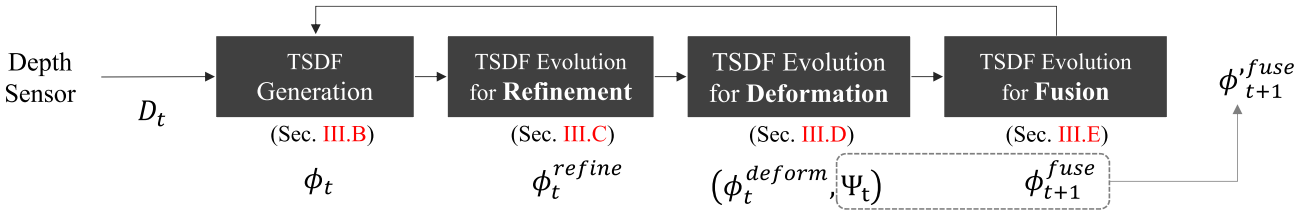


Fig. 3. An overview pipeline of the proposed reconstruction framework from a single depth sensor stream, where a live frame visualization ϕ'^{fuse}_{t+1} is accomplished by finding a deformation field Ψ_t to the canonical frame for a given depth frame D_t. The raw TSDF approximated using the conventional TSDF construction method is incomplete and has a lot of noise. TSDF values are accurately recovered in a few iterations through the refinement procedure, facilitating a more robust deformation procedure. The subsequent level-set evolution for the fusion procedure enables the framework to work reliably and accurately over time.

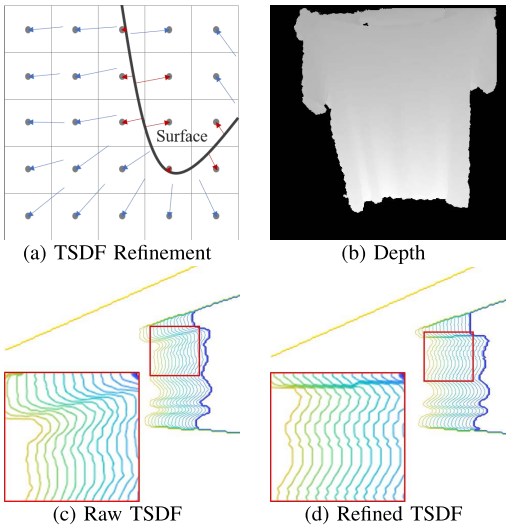


Fig. 4. (a) TSDF refinement procedure visualization and (b)−(d) iso-surface visualizations before and after the refinement procedure on "*Shirt*" in the VolumeDeform dataset [18]. The raw TSDF shows uneven and perspective biased iso-surface lines. In the refinement procedure, TSDF values are propagated iteratively from the normal direction of the object surface based on the property of the TSDF approximation that the point nearer to the object surface has a more reliable TSDF value and gradient. The refinement procedure efficiently improves TSDF representations while preserving the surface details in a few iterations.

an object's surface are from the depth sensor [1], [2] can be reliable. They are always close to zero regardless of the projected ray, as depicted in Fig. 1. The proposed refinement method propagates TSDF values from the surface by iteratively replacing each TSDF voxel value with that obtained from the direction of the unit normal. In other words, every voxel updates its values from the opposite direction of the object's surface. The unit step distance of the propagation is determined depending on the voxel size v_s. The TSDF refinement evolution at iteration $i \geq 1$ is formulated as

$$\phi_t^{i+1}(\mathbf{x}) = \phi_t^i(\mathbf{x} - sgn(\phi_t(\mathbf{x})) \cdot \mathbf{n}) + v_s \tag{4}$$

where $\mathbf{n} = \frac{\nabla\phi_t}{|\nabla\phi_t|}$. The refinement starts with the raw TSDF, i.e., $\phi_t^1(\mathbf{x}) = \phi_t(\mathbf{x})$.

In an implicit surface representation, the surface normal $\mathbf{n}$ in (4) can be calculated as the normalized gradient [54]. The sign operator determines whether the voxel is inside or outside the surface. When it is outside the surface, it obtains the value from the negative normal direction. When it is inside the surface, the normal direction is toward the surface. Because the method propagates the reliable values from the surface using the unit distance for every iteration, it is sufficient to repeat the propagation until TSDF truncated regions are achieved. Thus, the maximum number of iterations required for TSDF refinement is equal to or less than the magnitude of the truncated margin τ in the voxel unit. In practice, 3 to 5 voxels are used for τ in most of the previous studies [3], [22], [39], enabling TSDF values to be recovered in a few iterations by the proposed refinement method.

Voxels near the surface, whose distance from the surface is less than the voxel size v_s, need to be manipulated more precisely to preserve the object surface positions while improving the TSDF representation. Therefore, for a voxel whose distance from the surface is less than v_s, it uses the value on the surface via trilinear interpolation for the propagation, as described using the red dotted arrows in Fig. 4 (a). Thus, the formulation in (4) is revised as

$$\phi_t^{i+1}(\mathbf{x}) = \phi_t^i(\mathbf{x} - sgn(\phi_t(\mathbf{x})) \cdot \tilde{\mathbf{n}}) + \tilde{v}_s(\mathbf{x}) \tag{5}$$

where $\tilde{v}_s(\mathbf{x}) = \min(v_s, |\phi_t^i(\mathbf{x})|)$ and $\tilde{\mathbf{n}} = \frac{\nabla\phi_t}{|\nabla\phi_t|}\frac{\tilde{v}_s}{v_s}$.

Figs. 4 (b), (c), and (d) show an example of the proposed refinement evolution, where 3D iso-surface lines of the surface are visualized. The raw TSDF values are much affected by the captured or projected direction. In other words, the iso-surface lines are perspective biased since TSDF values are approximated along the perspective direction. In addition, iso-surface line intervals far from the object surface are significantly uneven, showing the object surface is not sufficiently represented by the implicit representation. The proposed refinement procedure improves the raw TSDF in a few iterations efficiently by propagating the TSDF values from the surface.

It is essential to calculate reliable gradients to obtain accurate surface normal directions because the TSDF values from a depth sensor are incomplete and noisy. From the definition of the finite difference, three types of gradients can be considered for the gradient operator: forward, backward, and central differences. The three types of gradients along the x coordinate can be defined, respectively, as

$$\nabla_x\phi_t^+ = \phi_t(x+1, y, z) - \phi_t(x, y, z), \tag{6}$$

$$\nabla_x\phi_t^- = \phi_t(x, y, z) - \phi_t(x-1, y, z), \tag{7}$$

$$\nabla_x\phi_t^0 = \frac{\nabla_x\phi_t^+ + \nabla_x\phi_t^-}{2}. \tag{8}$$

In many computer vision and image processing algorithms, the central difference is preferred since it yields approximations more accurate and robust to noise. However, it is difficult to manipulate discontinuous and invalid values efficiently of raw TSDFs using the central difference.

Inherently, TSDF values near the surface are more reliable than values far from the surface. In addition, as TSDF values are propagated from the surface in the proposed evolution for TSDF refinement, values nearer to the surface become more reliable. Given these TSDF values, a reliable gradient can be measured from the surface direction, i. e., the propagated direction.

Thus, the gradient along x-coordinates is defined for our propagation as

$$\nabla_x \phi_t = \begin{cases} \nabla_x \phi_t^+, & \text{if } \phi_t \geq 0,\ \nabla_x \phi_t^+ \leq 0 \\ & \text{or } \phi_t < 0,\ \nabla_x \phi_t^+ \geq 0 \\ \nabla_x \phi_t^-, & \text{if } \phi_t < 0,\ \nabla_x \phi_t^- \geq 0 \\ & \text{or } \phi_t \geq 0,\ \nabla_x \phi_t^- \leq 0 \\ 0. & \text{otherwise} \end{cases} \tag{9}$$

For reliability, saddle points are defined as zero gradients. The formulation in (9) is also used for gradients along y-coordinates.

A different scheme is used for calculating gradients along z-coordinates because we have a significant prior along the depth direction. For raw TSDFs from a calibrated depth sensor (i.e., the camera intrinsic is known), the surface normal is always toward the negative z-axis. Therefore, TSDF gradients along z-coordinates can be defined more simply and accurately as

$$\nabla_z \phi_t = \begin{cases} \nabla_z \phi_t^-, & \phi_t \geq 0 \\ \nabla_z \phi_t^+. & \phi_t < 0 \end{cases} \tag{10}$$

The gradient measurements proposed in (9) and (10) allow obtaining the gradients from the surface direction. In particular, the measurement in (9) calculates the gradients from the surface direction along x- and y-coordinates by seeking the increasing or decreasing direction of the gradient flow. Meanwhile, the measurement in (10) calculates the z-coordinate gradient using capture direction priors of a depth sensor. The measurements enable our refinement method to update the TSDF voxels from near-surface towards far-from-surface fully parallel without a specific order.

Fig. 5 depicts the magnitude of the surface gradient and the direction of the surface normal at the beginning, refinement iterations $i = 2$ and 5. The TSDF and gradient values of the truncated interval (± 5 voxels) of the surface are visualized. The refinement procedure accurately corrects invalid gradients in raw TSDFs while preserving the object surface by propagating the gradients from the surface direction. The propagation restores smooth and complete gradients with uneven and noisy gradients within a few numbers of iterations, i.e., the truncated voxel size ($i = 5$). When using the central difference to calculate TSDF gradients for the refinement procedure, the propagation does not sufficiently improve uneven gradients because both invalid and valid gradients are used for propagation regardless of the surface direction. In contrast, using the more reliable gradients measured from the propagated direction enables the refinement method to improve raw TSDFs more accurately.

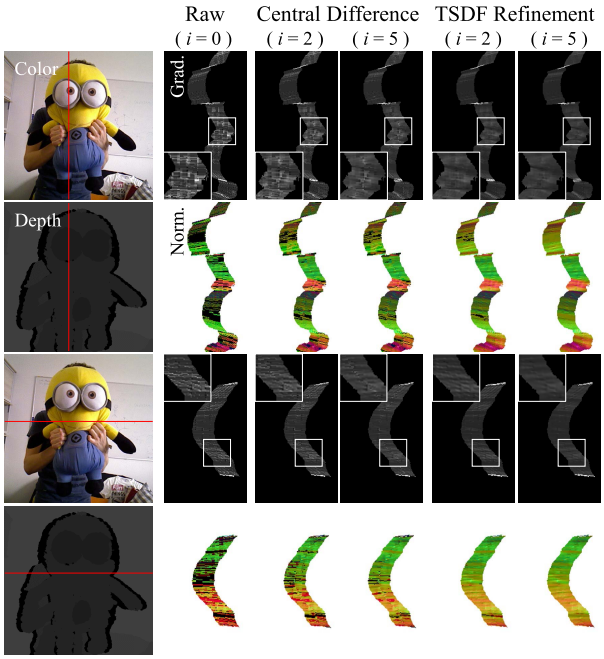


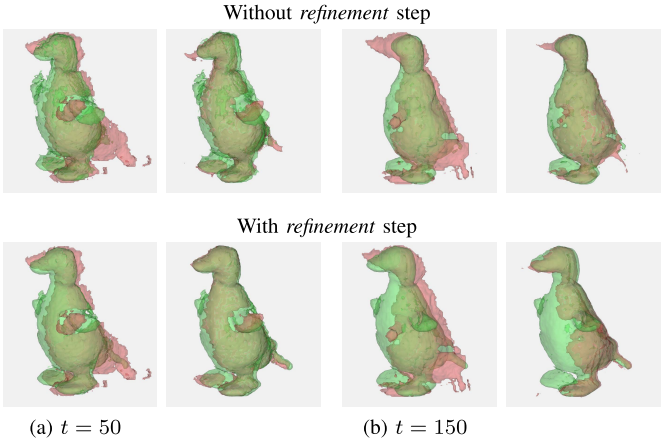
Fig. 5. The surface gradient and normal visualizations of the truncated interval (± 5 voxels) at the refinement iterations $i = 0, 2$, and 5 on "*Minion*" in the VolumeDeform dataset [18]. The raw TSDF values approximated from the depth map are incomplete and erroneous ($i = 0$). The proposed refinement method improves uneven TSDF values and gradients of the raw TSDF in a few iterations by propagating the TSDF values from the surface direction. The method measures the surface direction toward each voxel using the gradients nearer to the surface because (1) the raw TSDF has more accurate values and gradients nearer to the surface initially and (2) the refinement method propagates more reliable TSDF values from near the surface to far; thus, more accurate values can be obtained from the nearer direction over the refinement iterations. Compared with the refinement evolution using the surface direction estimation from the central difference, the proposed surface direction estimation efficiently supports the proposed evolution to refine the TSDF values more reliably and accurately.

D. TSDF Deformation

The deformation step is a procedure to find voxel-wise motion vectors to the canonical TSDF. We denote a vector warp field $\Psi = (U, V, W) \in \mathbb{R}^3$ that aligns incoming TSDFs at time t ($\phi_t(\mathbf{x})$) to the reference TSDF $\left(\phi_t^{fuse}(\mathbf{x})\right)$ in the canonical coordinates. In other words, a deformed TSDF $\phi_t^{deform}(\mathbf{x})$ is found in the deformation step, satisfying $\phi_t^{deform}(\mathbf{x}) = \phi_t(\mathbf{x} + \Psi) \simeq \phi_t^{fuse}(\mathbf{x})$.

Our framework mainly tackles the temporal reliability limitation of incremental reconstruction techniques using refinement and fusion propagation procedures to stably preserve TSDF representation in time in contrast to other relevant works that focused on the deformation step. Therefore, we use the simple variational formulation widely used to find a motion field, i.e., scene flow [70]–[74] for TSDF deformation. Thus, two energy terms are defined for non-rigid 3D reconstruction: data and smoothness terms as

$$E_{deform}(\Psi) = E_{data}(\Psi) + \omega_{smooth} E_{smooth}(\Psi) \tag{11}$$



(a) $t = 50$ (b) $t = 150$

Fig. 6. Comparisons with and without the refinement step before the deformation step performed on "*Duck Loop*" in the deformable 3D reconstruction dataset [3]. In each pair of images, the source object (colored in red) is non-rigidly registered to the target object (colored in green) in the TSDF from the initial state in the first of the pair images. Although the misalignment error seems to be insignificant at $t = 50$ for the deformation without refinement, tit increases over time and can significantly affect the reconstruction pipeline's accuracy, as seen in the deformation at $t = 150$. It is shown that the refinement evolution plays a significant role in the accurate and reliable deformation of TSDFs in the incremental pipeline while preventing error accumulation in TSDFs over time.

where ω_{smooth} is a constant that balances the smoothness of motions. The data term and smoothness term can be defined, respectively, as

$$E_{data}(\Psi) = \frac{1}{2}\sum_{\mathbf{x}}\left(\phi_t(\mathbf{x}+\Psi) - \phi_t^{fuse}(\mathbf{x})\right)^2, \tag{12}$$

$$E_{smooth}(\Psi) = \frac{1}{2}\sum_{\mathbf{x}}\left(|\nabla U|^2 + |\nabla V|^2 + |\nabla W|^2\right). \tag{13}$$

The vector warp field is obtained by updating the field iteratively using the variational derivative of the energy in (11). The formulation at iteration $k \geq 1$ is represented as:

$$\Psi_t^{k+1} = \Psi_t^k + \alpha_{deform}\nabla E_{deform}(\Psi_t) \tag{14}$$

where $\nabla E_{deform} = \nabla E_{data} + \omega_{smooth}\nabla E_{smooth}$ and α_{deform} is the step size of gradient descent minimization. For an initial warp at the first depth frame $\Psi_1^1(\mathbf{x})$, all motion vectors are set as zeros. For initial warps at the following frames $\Psi_t^1(\mathbf{x})$ ($t \geq 2$), the final warp at the previous frame is used. In our implementation, the iteration ends when the maximum vector update in (14) reaches below a threshold of 0.1 *mm*. Using the calculus of variations, the derivatives of energies in (12) and (13) are obtained, respectively, as

$$\nabla E_{data}(\Psi) = \left(\phi_t(\mathbf{x}+\Psi) - \phi_{ref}(\mathbf{x})\right)\nabla\phi_t(\mathbf{x}+\Psi), \tag{15}$$

$$\nabla E_{smooth}(\Psi) = -(\Delta U, \Delta V, \Delta W). \tag{16}$$

Even with the simple formulation for the deformation in (11), the proposed refinement method provides reliable gradient measurements, enabling a source TSDF to be accurately registered to a target TSDF. Fig. 6 depicts the benefit of the refinement procedure in the deformation step, where a source object (colored in red) is deformed to a target object (colored in green). As the variational method, including other optimization schemes, finds the optimal solution of the energy in (11) using TSDF gradients, the more accurate and reliable gradients improve the accuracy of the deformation step.

E. Level-Set Evolution for the Fusion

As described in Fig. 3, a raw TSDF representation $\phi_t(\mathbf{x})$ at time t in (2) is improved in the refinement step using (5), producing a refined TSDF $\phi_t^{refine}(\mathbf{x})$. In the deformation step, the refined TSDF $\phi_t^{refine}(\mathbf{x})$ subsequently register non-rigidly the reference TSDF $\phi_t^{fuse}(\mathbf{x})$, producing a aligned TSDF $\omega_t^{deform}(\mathbf{x})$. The reference TSDF $\phi_t^{fuse}(\mathbf{x})$ is the accumulation of the TSDFs from the first to the previous frame $t-1$ in the canonical coordinates. In general, the coordinates of the first frame [3], [22] or the keyframe [39] is chosen as the reference frame. In our implementation, the first frame is used as the reference frame.

The aligned TSDF $\omega_t^{deform}(\mathbf{x})$ is combined with the reference TSDF, producing the reference TSDF of the following iteration $\phi_{t+1}^{fuse}(\mathbf{x})$. The procedure is denoted as *TSDF fusion*. In the incremental reconstruction, the fusion step is a significant procedure to integrate a target object from multiple frames over time. The fusion accuracy considerably depends on the quality of TSDFs from the deformation step. Therefore, for a more reliable fusion procedure, the TSDF refinement step is proposed in Sec. III-C to improve the accuracy of the deformation step. Nevertheless, the accuracy of TSDF fusion significantly affects the overall reliability of the reconstruction pipeline in the subsequent frames. We propose a level-set fusion evolution for the pipeline reconstruction pipeline to make the reconstruction more robust in time.

The TSDF fusion can be represented by the following formulations using the weighted average scheme [27] as

$$\omega_{t+1}^{fuse}(\mathbf{x}) = \omega_t^{fuse}(\mathbf{x}) + \omega_t^{deform}(\mathbf{x}), \tag{17}$$

$$\phi_{t+1}^{fuse}(\mathbf{x}) = \frac{\omega_t^{fuse}(\mathbf{x})\cdot\phi_t^{fuse}(\mathbf{x}) + \omega_t^{deform}(\mathbf{x})\cdot\phi_t^{deform}(\mathbf{x})}{\omega_{t+1}^{fuse}(\mathbf{x})} \tag{18}$$

where $\omega_t^{fuse}(\mathbf{x})$ and $\omega_t^{deform}(\mathbf{x})$ are the TSDF weights corresponding to $\phi_t^{fuse}(\mathbf{x})$ and $\phi_t^{deform}(\mathbf{x})$, respectively, for the voxel $\mathbf{x}$ at the iteration t.

However, the non-rigid deformation of TSDFs does not preserve the magnitude of TSDF gradients. In addition, the TSDF gradients tend to be unreliable and irregular due to noise and misalignment. The weighted average fusion scheme can mitigate the effect of those unreliable values on the fused TSDF by preventing rapid changes in TSDF values based on temporal statistics. Nevertheless, the weighted average scheme does not resolve but accumulates inaccurately represented values gradually, producing unreliable reconstruction results in time. The truncated weight [1], [2], [16] and the reference-biased weight [39] schemes have been introduced to decrease artifacts caused by the fusion procedure. In the truncated weight scheme, the maximum magnitude of the TSDF weight in (18) is truncated with the predefined value ω_{max}, i.e., $\omega_{t+1}^{fuse}(\mathbf{x}) = \min(\omega_{t+1}^{fuse}(\mathbf{x}), \omega_{max})$. The truncated weight scheme can ensure the minimal influence of the incoming frame on the fused TSDF over time. The reference-biased

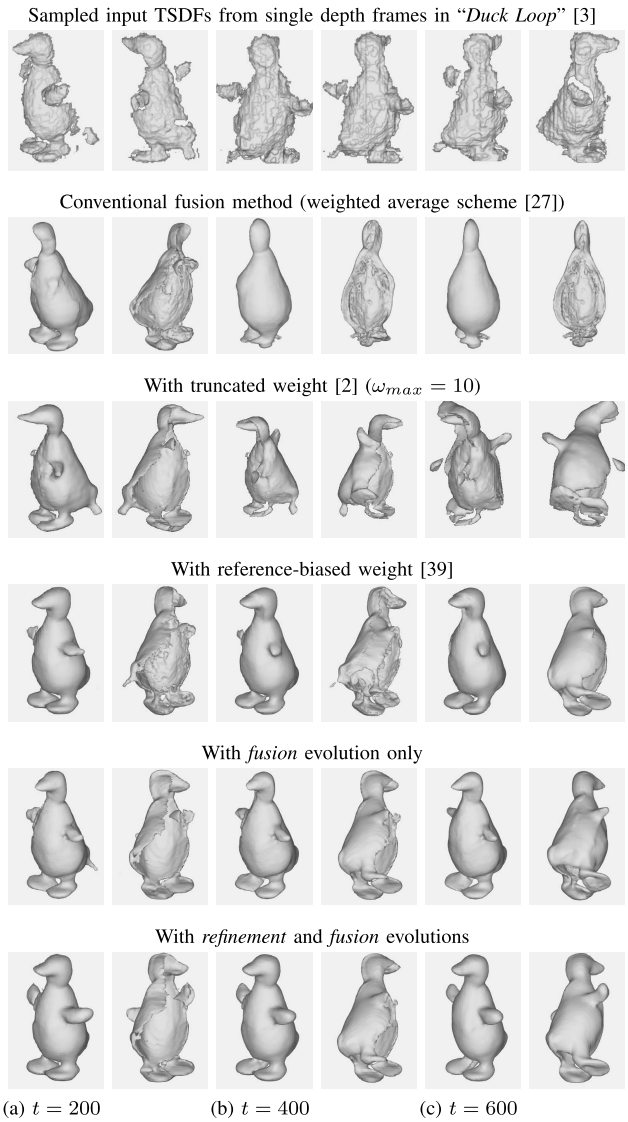


Fig. 7. Fusion comparisons using different fusion schemes on "*Duck Loop*" in the deformable 3D reconstruction dataset [3] (first row). The 3D reconstructed results in the canonical coordinates at frame $t = 200, 400$, and 600 are visualized. It can be seen that the fusion evolution facilitates reliable 3D reconstruction from a depth sequence of a dynamically moving object. Compared to other schemes [2], [27], [39], the proposed fusion evolution reconstructs a target surface more accurately and incrementally. The combination of refinement and fusion schemes in the reconstruction pipeline significantly increases the reliability of the pipeline, enabling precise reconstruction over time.

weight scheme provides linearly lower weights depending on the distance from the reference TSDF, i.e., $\tilde{\omega}_t^{deform}(\mathbf{x}) = \left(1 - \frac{\left|\phi_t^{fuse}(\mathbf{x}) - \phi_t^{deform}(\mathbf{x})\right|}{2}\right) \cdot \omega_t^{deform}(\mathbf{x})$. Although these fusion schemes help to manipulate the change rate of the canonical TSDF efficiently, they cannot sufficiently maintain the TSDF representation in time.

Fig. 7 shows reconstruction results in the canonical coordinates performed on a depth stream of a dynamically moving object. Owing to misalignment in the weighted average scheme, artifacts worsen tracking and reconstruction performance gradually in time. The truncated weight scheme does not cope well with a dynamic object as it accelerates the propagation of misalignment errors into the reference TSDF. The reference-biased weight scheme helps 3D shapes in the canonical coordinates to be more preserved against erroneous deformation than the other schemes; however, it does not sufficiently help TSDFs to accurately deform and reflect a target shape from a depth stream.

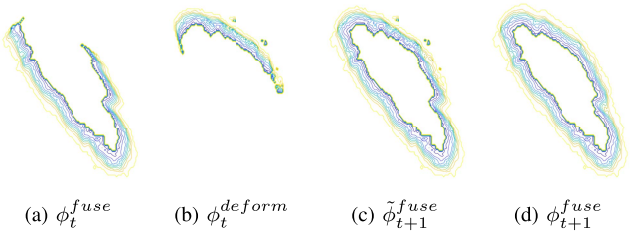


Fig. 8. The iso-surface visualizations of TSDFs that are incorporated in the TSDF fusion procedure. The reference TSDF of the previous frame ϕ_t^{fuse} (a) and the deformed TSDF of the current frame ϕ_t^{deform} (b) is fused using the conventional fusion scheme [27], yielding the raw fused TSDF $\tilde{\phi}_{t+1}^{fuse}$ (c), which contains noisy, distorted, and irregularly distanced values. The proposed fusion evolution addresses the problem (noise, distortion, etc.) by propagating ϕ_t^{fuse} along the gradient flow toward $\tilde{\phi}_{t+1}^{fuse}$ while regularizing the TSDF distance. The efficiently regularized TSDF through the fusion propagation ϕ_{t+1}^{fuse} (d) increases the temporal reliability of the reconstruction pipeline.

In the following, we denote the raw fused TSDF by the formulation in (18) as $\tilde{\phi}_{t+1}^{fuse}$ at the current frame and the output from the proposed TSDF fusion evolution as ϕ_{t+1}^{fuse} for clarity. In the proposed evolution for TSDF fusion, the reference TSDF at the previous frame (ϕ_{t+1}^{fuse}) is propagated toward the raw fused TSDF $\tilde{\phi}_{t+1}^{fuse}$ at the current frame, consequently producing the reference TSDF at the current frame (ϕ_{t+1}^{fuse}). As depicted in Fig. 8, the evolution prevents distinct value changes caused by noise and misalignment by propagating gradients of the source TSDF ϕ_t^{fuse} toward the fused TSDF ϕ_{t+1}^{fuse}. Besides, TSDF gradients are regularized with a unit scale during the evolution. In other words, the evolution recovers TSDFs by following the property of the implicit surface representation. Thus, we define the variational formulation for TSDF fusion, which is composed of two energy definitions: data (E_{data}) and regularization (E_{dist}) terms. The TSDF fusion evolution energy is defined as

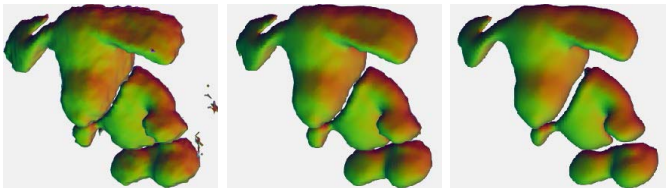
$$E_{fuse}(\phi_t) = \sum_{\mathbf{x}} E_{data}(\phi_t) + \omega_{dist} E_{dist}(\phi_t) \tag{19}$$

where $E_{data}(\phi_t(\mathbf{x})) = \frac{1}{2}\left(\tilde{\phi}_{t+1}^{fuse}(\mathbf{x}) - \phi_t(\mathbf{x})\right)^2$ and $\omega_{dist} > 0$ is a constant that balances propagation and regularization. We use the regularization energy for an implicit function proposed by Li *et al.* [51] to preserve the gradient magnitudes. The regularization term is defined as

$$E_{dist}(\phi_t) = \int p(|\nabla \phi_t|) d\mathbf{x} \tag{20}$$

where

$$p(s) = \begin{cases} \frac{1}{(2\pi)^2}(1 - \cos(2\pi s)), & \text{if } s \leq 1 \\ \frac{1}{2}(s-1)^2. & \text{if } s > 1 \end{cases}$$



(a) $\omega_{dist} = 0.005$ (b) $\omega_{dist} = 0.02$ (c) $\omega_{dist} = 0.1$

Fig. 9. Fusion visualizations colored with surface normals according to the parameter ω_{dist} on "*Snoopy*" in the deformable 3D reconstruction dataset [3]. The distance regularization aids in accurately representing a target surface by preserving the TSDF property. However, large values can result in overly smooth surfaces.

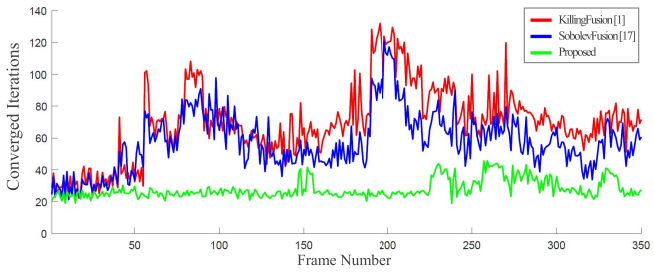


Fig. 10. Convergence iteration comparisons of variational approaches for the TSDF reconstruction on a single-view RGB-D stream of "*Alex*" in the deformable 3D reconstruction dataset [3]. The reliable TSDF manipulation of the proposed method significantly decreases the number of iterations that converge.

The constant ω_{dist} controls the magnitude of the TSDF distance regularization. The regularization term plays a significant role maintaining the signed distance property (i.e., the unit magnitude) in evolving TSDFs. However, extremely strong regularization can cause TSDFs to evolve into representing over-smooth surfaces, as described in Fig.9. Therefore, a proper value needs to be selected to balance the temporal reliability and high-frequency details.

The energy in (19) is minimized by iteratively updating TSDF voxel values using the variational derivative. The formulation at iteration $j \geq 1$ is represented as

$$\phi_t^{j+1}(\mathbf{x}) = \phi_t^j(\mathbf{x}) + \alpha_{fuse} \nabla E_{fuse} \left(\phi_t^j(\mathbf{x}) \right) \tag{21}$$

where $\nabla E_{fuse} = \nabla E_{data} + \mu \nabla E_{dist}$ and α_{fuse} is a step size of gradient descent minimization. The fusion propagation in (21) starts from the reference TSDF (i.e., $\phi_t^1(\mathbf{x}) = \phi_t^{fuse}(\mathbf{x})$) and iterates until it converges. In our implementation, the iteration ends when the maximum magnitude of the gradient over voxels in (21) reaches below a threshold of 0.1 mm. The average number of iterations in our experiments is 19.3.

The derivative of the data term ∇E_{data} is obtained using the standard calculus of variations [54] as

$$\nabla E_{data}(\mathbf{x}) = \left(\tilde{\phi}_{t+1}^{fuse}(\mathbf{x}) - \phi_t^j(\mathbf{x}) \right) \left| \nabla \phi_t^j(\mathbf{x}) \right|. \tag{22}$$

The derivative of the distance regularization term ∇E_{dist} in (20) is represented as

$$\nabla E_{dist} = \operatorname{div} \left(d_p \left(\left| \nabla \phi_t^j \right| \right) \nabla \phi_t^j \right) \tag{23}$$

where

$$d_p(s) = \begin{cases} \dfrac{1}{2\pi} \sin(2\pi s), & \text{if } s \leq 1 \\ s - 1, & \text{if } s > 1 \end{cases}$$

and $\operatorname{div}(\cdot)$ is the divergence operator. The regularizer in (23) decreases $|\nabla\phi|$ when $|\nabla\phi| > 1$, whereas increases $|\nabla\phi|$ when $\frac{1}{2} < |\nabla\phi| < 1$. It efficiently handles noise and outliers by further decreasing $|\nabla\phi|$ when $|\nabla\phi| < \frac{1}{2}$.

Fig. 2 and the last two rows of Fig. 7 show examples of the evolution of the TSDF fusion over time. While different fusion schemes can be useful in certain situations, it is necessary to preserve the reliable TSDF representation for accurate tracking and reconstruction from a depth stream temporally. In particular, the combination of the two evolution methods for the TSDF refinement and fusion enables the proposed framework to reliably and accurately deform TSDFs significantly over time without regularizing gradients during deformation [75], penalizing volume distortions [3], using gradient flow in the Sobolev space [22], or using feature terms in the color space [18], [39]. The performance comparisons of methods that used these techniques are presented in the experimental section.

IV. Experimental Results

In this section, we compared the proposed framework to state-of-the-art pipelines for 3D reconstructions using a single RGB-D camera. The evaluations were performed on public RGB-D datasets. For dynamic scenes, the VolumeDeform dataset [18] captured using *Asus Xtion PRO* and the deformable 3D reconstruction dataset [3] captured using *Microsoft Kinect v1* were used. The dataset in the work of Tsoli and Argyros [76] captured using *Microsoft Kinect v2* was used for topology-changing scenes. To demonstrate the effectiveness of the proposed method, we constructed RGB-D sequences using one of the latest RGB-D sensors, *Microsoft Kinect Azure*, to validate the reconstruction performance of the proposed framework on a rigid scene and sequences of more dynamic objects, such as a person in motion. In addition, the 3D sequence dataset in [40] was used to quantitatively evaluate the tracking feasibility of the proposed method in a dynamic scene. The entire sequences of Figs. 11, 14, and 16 shown in the experimental section are presented in *the Supplementary Material*. All the results in the experiments were obtained without any pre-computation or template model.

A. Implementation Details

The TSDF evolutions in Secs. III-C, III-D, and III-E, including the generation in Sec. III-B, are fully parallelizable for each voxel. We use one Nvidia RTX 2080 Ti GPU for testing in our experiments. We used the bounding volume of $150 \times 150 \times 150$ for every test. Depending on the target object size, we vary the voxel size 3-10 mm to fit the object into the bounding volume. In our implementation, the parameters of our framework are set as $\omega_{smooth} = 0.3$ and $\alpha_{deform} = 0.2$ for deformation and $\omega_{dist} = 0.02$ and $\alpha_{fuse} = 0.2$ for fusion propagation. As we set the truncated margin τ to five times the voxel size v_s, the number of iterations for the refinement is set to 5. The refinement, fusion, and deformation procedures cost 4 ms, 9 ms, and 19 ms on

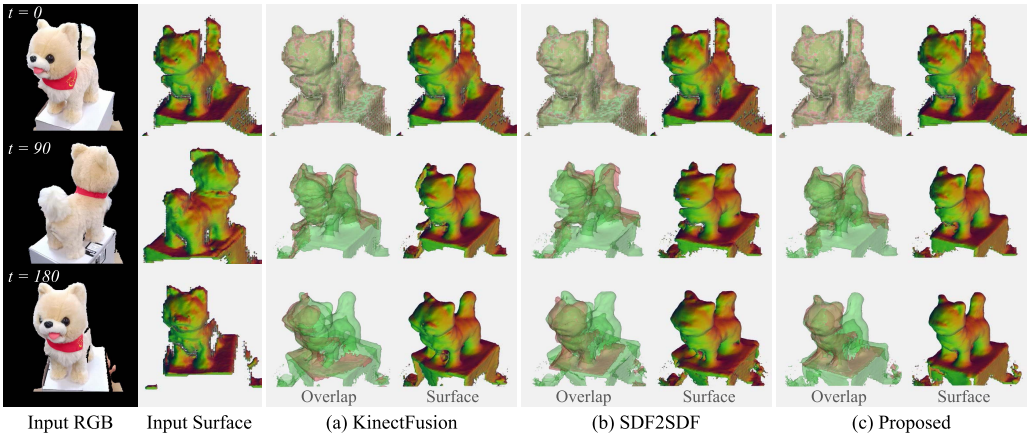


Fig. 11. Qualitative comparisons on the rigid scene at frame $t = 0, 90$, and 180. The surface overlaps between the canonical (colored in green) and wrapped TSDFs (colored in red) and the reconstruction results obtained from (a) KinectFusion [1], [2], (b) SDF2SDF [12], and (c) the proposed method.

average, respectively. To achieve 30 frames per second more consistently, our framework pipeline is performed in one-frame delayed in real-time.

B. Convergence

The number of iterations for the proposed deformation procedure to converge is evaluated and compared with those of two state-of-the-art variational deformation schemes for TSDF reconstruction from a single stream: KillingFusion [3] and SobolevFusion [22]. KillingFusion uses an approximately Killing vector field constraint [77] to penalize volume distortions during deformation. In SobolevFusion, the gradient flow of TSDFs is calculated by projecting TSDF gradients into the Sobolev space [78] to evolve TSDFs in a coarse-to-fine manner. We assume that the optimization procedure converges when the maximum vector update in (14) reaches below a threshold 0.1 *mm*. Fig. 10 depicts the convergence iterations for the variational schemes on the large-motion sequence, "*Alex*" in the deformable 3D reconstruction dataset [3]. The average iterations and standard deviations of the KillingFusion, SobolevFusion, and proposed measure 69.225±22.768, 57.481±12.250, and 27.832±5.693, respectively. SobolevFusion shows a lower number of iterations than KillingFusion because it uses scalable gradients calculated in the Sobolev space. The proposed method significantly decreases the number of iterations by reliably maintaining the signed distance representation. A low standard deviation indicates that the proposed pipeline can rapidly operate with temporal stability.

C. Experiments on Rigid Scene

To verify the proposed evolution methods, we evaluate the reconstruction performance on the rigid scene. The comparisons are conducted with two rigid TSDF reconstruction methods: KinectFusion [1], [2] and SDF2SDF [12]. KinectFusion finds a rigid transform between the iso-surface of the TSDF and a given frame depth using ICP. SDF2SDF is a variational method that uses the TSDF gradients to find a rigid transform. For fair comparison with the rigid reconstruction methods, we update all the TSDF voxels by an average value of the deformation derivative in (15) over voxels to transform the TSDF rigidly. The smoothing term in (16) does not affect because the laplacian of the rigid wrap is zero. Fig. 11 shows reconstructions in the canonical frame and surface overlaps between the canonical and wrapped TSDFs at initial and two different intermediate times. Although these lead to minor misalignment in a pair of consecutive frames, the accumulated misalignment error over time can cause large artifacts. The reconstruction results of KinectFusion show an example of the accumulated artifact, a dog that has a couple of heads. Compared to KinectFusion that minimizes the surface distance, SDF2SDF minimizes the TSDF difference. Thus, every valid value of the TSDF is involved in the optimization procedure of SDF2SDF, enabling more accurate results than KinectFusion. Nevertheless, the results show that noisy depths and accumulated errors over time significantly affect the alignment and reconstruction of SDF2SDF. The proposed method shows the most accurate and reliable alignment and reconstruction results by handling errors over time, demonstrating the effectiveness of the proposed TSDF evolution method.

D. Experiments on Dynamic Scene

We compare the accuracy of our pipeline with three state-of-the-art TSDF reconstruction methods: KillingFusion [3], SobolevFusion [22], and SurfelWarp [17]. SurfelWarp improves DynamicFusion [16] using surfel based representation. Similar to the proposed deformation step, SobolevFusion and KillingFusion use the variational formulation for deformation. Figs. 12 and 13 show qualitative and quantitative non-rigid reconstruction results on single-view streams of VolumeDeform [18] and the deformable 3D reconstruction [3] datasets.

KillingFusion has the lowest geometric accuracy among the methods on both datasets. The variational level-set approach, used in the KillingFusion, SobolevFusion, and the proposed method, tracks the TSDF volume voxel-wisely, enabling a sub-voxel level accuracy deformation. However, variational optimization is much more vulnerable to error and noise of

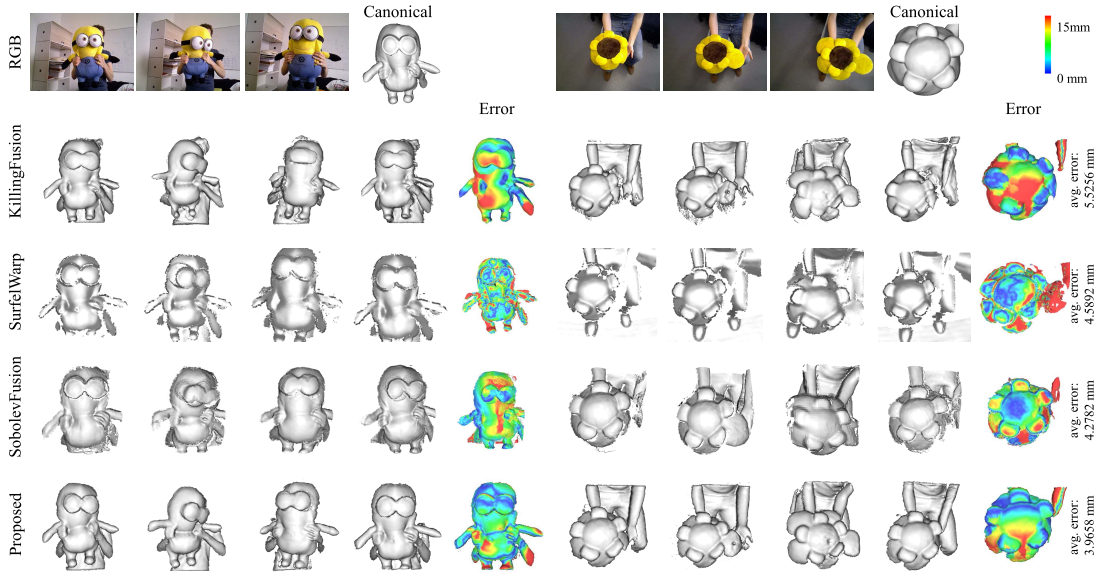


Fig. 12. Qualitative and quantitative non-rigid reconstruction results on single-view RGB-D streams of "*Minion*" and "*Sunflower*" in the VolumeDeform [18] dataset.

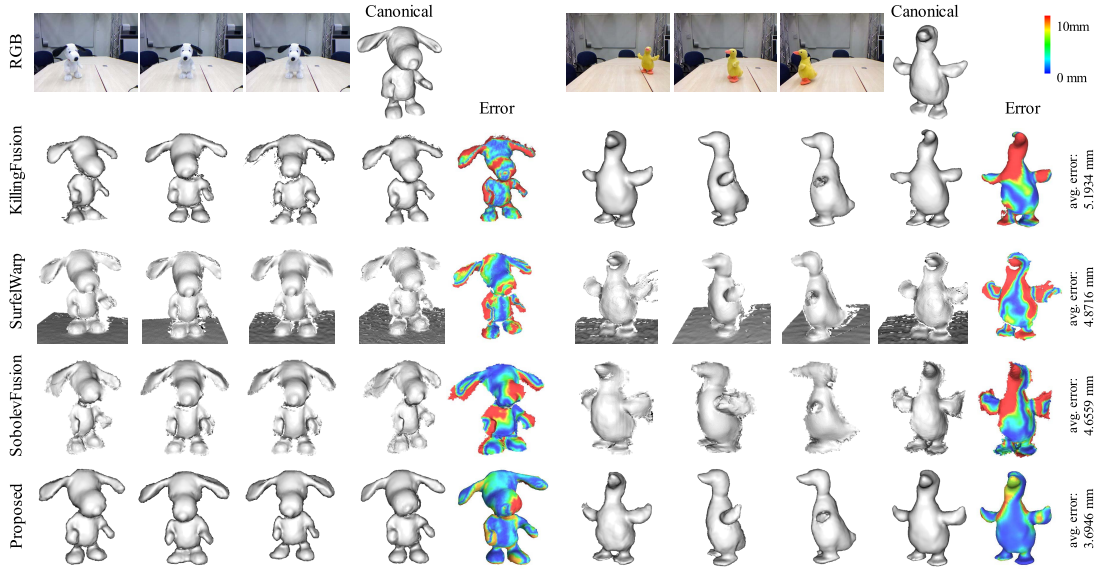
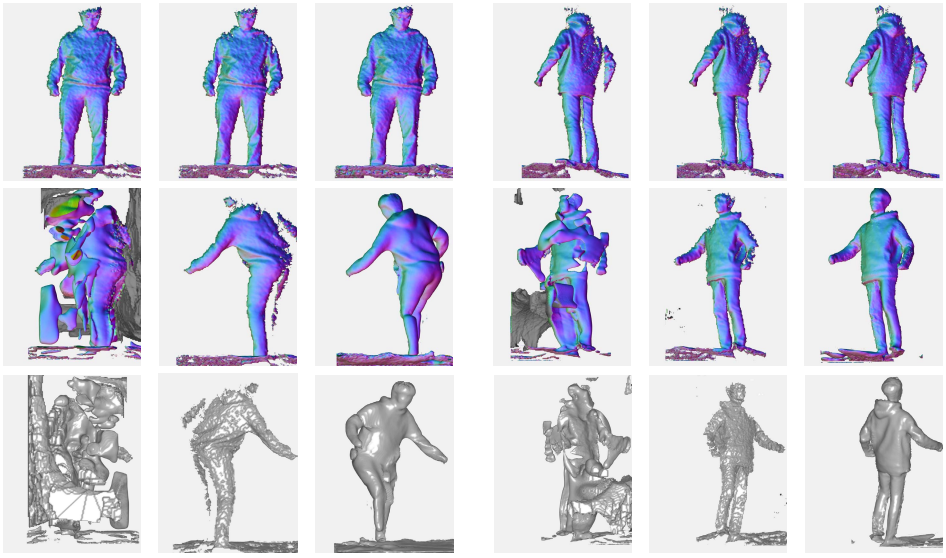


Fig. 13. Qualitative and quantitative non-rigid reconstruction results on single-view RGB-D streams of "*Snoopy*" and "*Duck*" in the deformable 3D reconstruction dataset [3] dataset.

TSDFs because voxel-wise gradient propagation is prone to get stuck in local minima. SurfelWarp uses an embedded deformation (ED) graphs [79] to parameterize 3D surface deformation. The ED graph transforms a 3D surface smoothly and densely over space using a sparsely sampled point set. The deformation using sparse points prevents geometric optimization from getting stuck into local minima, achieving better accuracy on average than KillingFusion. Despite the better convergence of ED parameterization, it is difficult to obtain the fine-scale details of the 3D surface from a sparsely sampled set of transformation basis functions. SobolevFusion shows better results than KillingFusion in our experiments by hierarchical propagation using the gradient in Sobolev space, similar to results as reported in [22]. SobolevFusion achieves better accuracy than SurfelWarp by efficiently coping with noisy gradients using a coarse-to-fine evolution in the Sobolev space. Nevertheless, SobolevFusion and KillingFusion innately suffer from the instability and incompleteness of TSDFs caused by their generation and fusion, accumulating TSDF errors in time. Gradient measurement in an erroneous TSDF can make the overall pipeline unreliable. The results demonstrate the proposed framework's ability to reconstruct 3D geometry more accurately and robustly. Our framework achieves better accuracy and reliability. It has been demonstrated that maintaining the TSDF property in time significantly increases reconstruction accuracy, enabling stable and reliable tracking and reconstruction from a depth sensor.



(a) KinectFusion (b) SobolevFusion (c) Proposed (d) KinectFusion (e) SobolevFusion (f) Proposed

Fig. 14. Qualitative comparisons of dynamic reconstructions on single streams in the dataset captured using *Microsoft Kinect Azure* using (a), (d) KinectFusion [1], [2], (b), (e) SobolevFusion [22] and (c), (f) the proposed method, respectively. The first and second row visualize the reconstruction results colored with the surface normal at the frame numbers 50 and 300, respectively. The last row shows back-face visualizations corresponding to the second row.

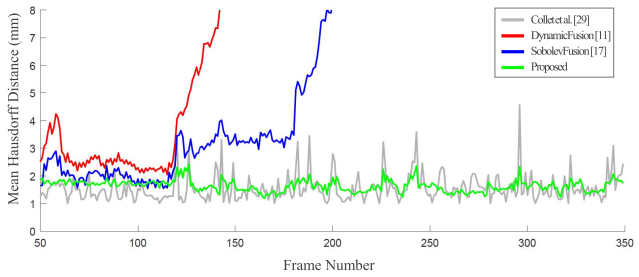


Fig. 15. Quantitative comparison on the multiview stream "*Break Dancers*" in the dataset of [40]. The proposed framework performs reliably on the multiview sequence, showing a similar mean error to one of the state-of-the-art offline methods.

E. Experiments on Largely Dynamic Sequences

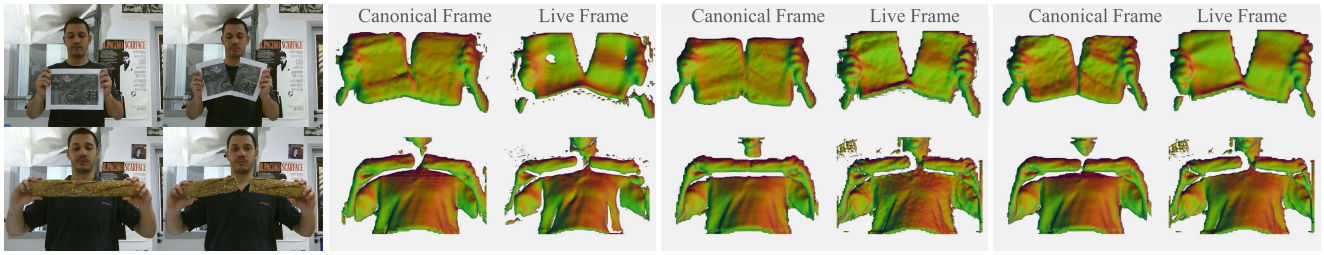
We validate the tracking and reconstruction performance of the proposed framework on the streams of more dynamic objects, such as a person in motion. The results are compared with SobolevFusion [22], which shows better performance than the other comparison methods in Sec. IV-D. In addition, the results of KinectFusion [1], [2] are compared with those of the proposed method to more clearly validate reconstruction performance between rigid and non-rigid techniques. Fig. 14 shows qualitative comparisons between the methods on single streams captured using *Microsoft Kinect Azure*. In the first two rows, the reconstructed shapes at frame numbers 50 and 300 colored with the surface normal are represented, respectively. The last row represents the back-side view corresponding to the reconstructed shape of the second row.

KinectFusion uses the surface location and gradient (normal) to match the correspondence to the new frame depth rather than fully using the TSDF gradients. Thus, KinectFusion is unable to sufficiently cope with noisy depths with uneven surface normals. In addition, it cannot handle object deformation at all, resulting in tracking failures and significant artifacts in its reconstructed shapes from dynamic streams. In the initial frames of the sequence (the first row), the difference between SobolevFusion and proposed is not significant. The shape reconstructed using the proposed framework is slightly less noisy. However, significant differences can be seen in sequence's intermediate frames (the second row). SobolevFusion reconstructs the 3D shape progressively in short-term sequences but hardly creates a complete 3D shape for the dynamic object. It is more clearly seen in the back-side view as described in the last row of Fig. 14. The proposed method reconstructs the entire object in the TSDF by tracking the dynamically moving object, whereas SobolevFusion eliminates the shape in invisible regions while tracking the visible region. The results demonstrate that the proposed framework enables the object to be tracked accurately in long-term sequences, which is especially beneficial for tracking dynamic objects.

To quantitatively validate reconstruction performance on the streams of dynamic objects, we measure the accuracy of the methods on the 3D dynamic sequence dataset in [40]. DynamicFusion [16] and SobolevFusion [22] are used for the comparison. In addition, we compared the proposed method with one of the state-of-the-art offline reconstruction methods [40]. Following the previous approach [39], we used the Hausdorff Distance [80] to measure the mean error against the ground truth mesh. DynamicFusion and SobolevFusion track a target object for the initial frames of the sequence; however, they fail to reliably reconstruct the target from the long-term dynamic sequence. In contrast, the proposed method stably tracks and reconstructs the object surface temporally, achieving results comparable to the offline method [40].

F. Experiments on Topology-Changing Scene

To validate the advantage of the proposed method on topology changes, we evaluate the reconstruction performance on the topology-changing dataset in Tsoli and Argyros [76].



(a) Input RGB (b) KillingFusion (c) SobolevFusion (d) Proposed

Fig. 16. Qualitative comparisons of non-rigid reconstruction results on "*Paper*" and "*Bread*" of the topology-changing dataset in Tsoli and Argyros [76]. (a) Input images of initial (canonical) and current frames, and reconstruction results in canonical and live-frame coordinates obtained from (b) KillingFusion [3], (c) SobolevFusion [22], and (d) the proposed method.

The comparisons are conducted with two variational methods: KillingFusion [3] and SobolevFusion [22]. These two methods are capable of handling topological changes in dynamic scenes by wrapping TSDF values voxel-wisely. We used 10 recent frames for fusing the TSDFs in the canonical space using (18) to clearly reflect the topological changes. Fig. 11 shows reconstruction results in canonical and live frames, respectively. The two images in Fig. 11 (a) depict initial and intermediate frames, respectively. The reconstructed TSDF from the initial frame is used as the canonical TSDF and updated over time by wrapping incoming TSDFs into the canonical TSDF. The reconstruction results in Figs. 11 (b), (c), and (d) are the surface visualizations of canonical and live-frame TSDFs at the intermediate frame time. The results show that all the comparison methods reflect topological changes over time into the canonical TSDF. Nevertheless, since the incoming depth is quite noisy, KillingFusion shows significant changes in the canonical TSDF by large misalignments, leading to instable live frame reconstruction results. The hierarchical gradient measurements in the Sobolev space allow SobolevFusion to reliably cope with topology changes. The proposed reconstruction framework shows the most clear-cut results thanks to accurate wrapping. The results demonstrate the significant advantage of the proposed method in topology changes. As a level-set-based deformation can handle topological changes, temporally manipulating reliable TSDF gradients through the proposed evolutions enables more accurate reconstruction of dynamic scenes under topology changes.

G. Limitation and Future Work

Although the proposed framework achieved qualified reconstructions in real-time frame rates, the TSDF representation of a 3D surface requires as many memories as the voxel volume size. It may restrict the applicability of the method depending on the maximum resolutions and computational powers of devices. However, a sparse [81] or hierarchical [82] structure representation can help solve the problem because most of values in a TSDF are truncated (i.e., background).

Moreover, depth-color correspondences obtained from *Microsoft Kinect Azure*, one of the latest off-the-shelf depth sensors, show incorrect and noisy results, in practice. Some examples of images, where RGB colors are mapped on the coordinates of depth map, are depicted in Fig. 17. Colors

Fig. 17. Matched depth-color correspondence examples on the depth map coordinates in our database captured using one of the latest depth sensors.

on object boundaries and rapidly moving components such as human hands are notoriously unreliable. Therefore, except when using clear depth maps obtained from the public database [83] or employing multi-depth cameras [38], [39], using color features for TSDF deformation does not help reconstruct more accurately for our cases. Nevertheless, we believe that the proposed framework contributes to the advancement of the incremental 3D surface reconstruction field by accounting for a reliable implicit function.

V. Conclusion

We presented a novel non-rigid 3D reconstruction pipeline from a single depth camera in real-time. An SDF approximated from a depth map is inaccurate except for values near a 3D surface, yielding inaccurate warping during the deformation step. The fusion procedure, a linear weighted summation between SDFs, generates artifacts, making 3D reconstruction unstable in time. We tackled the problems by introducing two significant gradient evolution methods for TSDF refinement and fusion with the reconstruction pipeline. Inaccurate TSDF values from a depth map can be recovered in a few iterations by propagating the TSDF values from the surface direction, leading to a more accurate deformation. The level-set evolution for the TSDF fusion helps the SDF to be manipulated reliably in time by propagating TSDF gradients with distance regularization. Our methods are fully parallelizable and can easily be used with previous reconstruction pipelines. We believe that our contribution provides a step toward real-time 3D surface reconstruction and tracking applications in everyday life.

References

[1] S. Izadi *et al.*, "KinectFusion: Real-time 3D reconstruction and interaction using a moving depth camera," in *Proc. 24th Annu. ACM Symp. User interface Softw. Technol. (UIST)*, 2011, pp. 559–568.

[2] R. A. Newcombe *et al.*, "KinectFusion: Real-time dense surface mapping and tracking," in *Proc. 10th IEEE Int. Symp. Mixed Augmented Reality*, Oct. 2011, pp. 127–136.

[3] M. Slavcheva, M. Baust, D. Cremers, and S. Ilic, "KillingFusion: Non-rigid 3D reconstruction without correspondences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1386–1395.

[4] J. Noraky and V. Sze, "Low power depth estimation of rigid objects for time-of-flight imaging," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 6, pp. 1524–1534, Jun. 2020.

[5] Z. Zhang, "Microsoft Kinect sensor and its effect," *IEEE Multimedia-Mag.*, vol. 19, no. 2, pp. 4–10, Feb. 2012.

[6] L. Keselman, J. I. Woodfill, A. Grunnet-Jepsen, and A. Bhowmik, "Intel(R) RealSense(TM) stereoscopic depth cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1–10.

[7] Z. Liu, J. Huang, J. Han, S. Bu, and J. Lv, "Human motion tracking by multiple RGBD cameras," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 9, pp. 2014–2027, Sep. 2017.

[8] C. Malleson, J.-Y. Guillemaut, and A. Hilton, "Hybrid modeling of non-rigid scenes from RGBD cameras," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 8, pp. 2391–2404, Aug. 2019.

[9] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. 3rd Int. Conf. 3-D Digit. Imag. Model.*, May 2001, pp. 145–152.

[10] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[11] M. Slavcheva, W. Kehl, N. Navab, and S. Ilic, "SDF-2-SDF registration for real-time 3D reconstruction from RGB-D data," *Int. J. Comput. Vis.*, vol. 126, no. 6, pp. 615–636, Jun. 2018.

[12] M. Slavcheva, W. Kehl, N. Navab, and S. Ilic, "SDF-2-SDF: Highly accurate 3D object reconstruction," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2016, pp. 680–696.

[13] F. Steinbrucker, J. Sturm, and D. Cremers, "Volumetric 3D mapping in real-time on a CPU," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 2021–2028.

[14] F. Steinbrucker, C. Kerl, and D. Cremers, "Large-scale multi-resolution surface reconstruction from RGB-D sequences," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3264–3271.

[15] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3D reconstruction at scale using voxel hashing," *ACM Trans. Graph.*, vol. 32, no. 6, pp. 1–11, Nov. 2013.

[16] R. A. Newcombe, D. Fox, and S. M. Seitz, "DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 343–352.

[17] W. Gao and R. Tedrake, "SurfelWarp: Efficient non-volumetric single view dynamic reconstruction," in *Proc. Robot., Sci. Syst. XIV*, Jun. 2018. [Online]. Available: http://www.roboticsproceedings.org/rss14/p29.html

[18] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger, "VolumeDeform: Real-time volumetric non-rigid reconstruction," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 362–379.

[19] K. Guo, F. Xu, T. Yu, X. Liu, Q. Dai, and Y. Liu, "Real-time geometry, albedo, and motion reconstruction using a single RGB-D camera," *ACM Trans. Graph.*, vol. 36, no. 3, pp. 1–13, Jul. 2017.

[20] T. Yu *et al.*, "DoubleFusion: Real-time capture of human performances with inner body shapes from a single depth sensor," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7287–7296.

[21] T. Yu *et al.*, "BodyFusion: Real-time capture of human motion and surface geometry using a single depth camera," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 910–919.

[22] M. Slavcheva, M. Baust, and S. Ilic, "SobolevFusion: 3D reconstruction of scenes undergoing free non-rigid motion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2646–2655.

[23] C. Li, Z. Zhao, and X. Guo, "ArticulatedFusion: Real-time reconstruction of motion, geometry and segmentation using a single depth camera," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 317–332.

[24] C. Li and X. Guo, "Topology-change-aware volumetric fusion for dynamic scene reconstruction," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2020, pp. 258–274.

[25] Z. Su *et al.*, "RobustFusion: Human volumetric capture with data-driven visual cues using a RGBD camera," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2020.

[26] Z. Li, T. Yu, Z. Zheng, K. Guo, and Y. Liu, "POSEFusion: Pose-guided selective fusion for single-view human volumetric capture," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021.

[27] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proc. 23rd Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 1996, pp. 303–312.

[28] S. Choi, A.-D. Nguyen, J. Kim, S. Ahn, and S. Lee, "Point cloud deformation for single image 3D reconstruction," in *Proc. ICIP*, Sep. 2019, pp. 2379–2383.

[29] D. Nguyen, S. Choi, W. Kim, and S. Lee, "GraphX-convolution for point cloud deformation in 2D-to-3D conversion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8628–8637.

[30] G. Turk and M. Levoy, "Zippered polygon meshes from range images," in *Proc. 21st Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 1994, pp. 311–318.

[31] M. Keller, D. Lefloch, M. Lambers, S. Izadi, T. Weyrich, and A. Kolb, "Real-time 3D reconstruction in dynamic scenes using point-based fusion," in *Proc. Int. Conf. 3D Vis.*, Jun. 2013, pp. 1–8.

[32] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "RGB-D mapping: Using Kinect-style depth cameras for dense 3D modeling of indoor environments," *Int. J. Robot. Res.*, vol. 31, no. 5, pp. 647–663, Apr. 2012.

[33] M. Kazhdan and H. Hoppe, "Screened Poisson surface reconstruction," *ACM Trans. Graph.*, vol. 32, no. 3, pp. 1–13, Jun. 2013.

[34] Q.-Y. Zhou and V. Koltun, "Dense scene reconstruction with points of interest," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 1–8, Jul. 2013.

[35] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Proc. Eurographics Symp. Geometry Process.*, vol. 7, Jun. 2006, pp. 61–70.

[36] T. Weise, T. Wismer, B. Leibe, and L. Van Gool, "In-hand scanning with online loop closure," in *Proc. IEEE 12th Int. Conf. Comput. Vis. Workshops, ICCV Workshops*, Sep. 2009, pp. 1630–1637.

[37] S. Rusinkiewicz, O. Hall-Holt, and M. Levoy, "Real-time 3D model acquisition," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 438–446, Jul. 2002.

[38] M. Dou *et al.*, "Motion2fusion: Real-time volumetric performance capture," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 1–16, Nov. 2017.

[39] M. Dou *et al.*, "Fusion4D: Real-time performance capture of challenging scenes," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 1–13, Jul. 2016.

[40] A. Collet *et al.*, "High-quality streamable free-viewpoint video," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 1–13, Jul. 2015.

[41] G. Ye, Y. Liu, N. Hasler, X. Ji, Q. Dai, and C. Theobalt, "Performance capture of interacting characters with handheld Kinects," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 828–841.

[42] J. Starck and A. Hilton, "Surface capture for performance-based animation," *IEEE Comput. Graph. Appl.*, vol. 27, no. 3, pp. 21–31, May 2007.

[43] H. Onizuka, Z. Hayirci, D. Thomas, A. Sugimoto, H. Uchiyama, and R.-I. Taniguchi, "TetraTSDF: 3D human reconstruction from a single image with a tetrahedral outer shell," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6011–6020.

[44] H. Li *et al.*, "Temporally coherent completion of dynamic shapes," *ACM Trans. Graph.*, vol. 31, no. 1, pp. 1–11, Jan. 2012.

[45] H. Li, B. Adams, L. J. Guibas, and M. Pauly, "Robust single-view geometry and motion reconstruction," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 1–10, Dec. 2009.

[46] H. Li, R. W. Sumner, and M. Pauly, "Global correspondence optimization for non-rigid registration of depth scans," *Comput. Graph. Forum*, vol. 27, no. 5, pp. 1421–1430, Jul. 2008.

[47] E. de Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun, "Performance capture from sparse multi-view video," in *Proc. ACM SIGGRAPH Papers (SIGGRAPH)*, 2008, pp. 1–10.

[48] C. Hernández, G. Vogiatzis, G. J. Brostow, B. Stenger, and R. Cipolla, "Non-rigid photometric stereo with colored lights," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.

[49] J. Kang and S. Lee, "A greedy pursuit approach for fitting 3D facial expression models," *IEEE Access*, vol. 8, pp. 192682–192692, 2020.

[50] S. Heo, H. Song, J. Kang, and S. Lee, "Local spherical harmonics for facial shape and albedo estimation," *IEEE Access*, vol. 8, pp. 177424–177436, 2020.

[51] C. Li, C. Xu, C. Gui, and M. D. Fox, "Distance regularized level set evolution and its application to image segmentation," *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3243–3254, Dec. 2010.

[52] H.-K. Zhao, T. Chan, B. Merriman, and S. Osher, "A variational level set approach to multiphase motion," *J. Comput. Phys.*, vol. 127, no. 1, pp. 179–195, Aug. 1996.

[53] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, Nov. 1988.

[54] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, vol. 153. New York, USA: Springer, 2006.

[55] S.-H. Lee, J. Kang, and S. Lee, "Enhanced particle-filtering framework for vessel segmentation and tracking," *Comput. Methods Programs Biomed.*, vol. 148, pp. 99–112, Sep. 2017.

[56] N. P. van Dijk, K. Maute, M. Langelaar, and F. van Keulen, "Level-set methods for structural topology optimization: A review," *Struct. Multidisciplinary Optim.*, vol. 48, no. 3, pp. 437–472, Sep. 2013.

[57] E. Angelini, Y. Jin, and A. Laine, "State of the art of level set methods in segmentation and registration of medical imaging modalities," in *Handbook of Biomedical Image Analysis*. Boston, MA, USA: Springer, 2005, pp. 47–101.
[58] B. Zitová and J. Flusser, "Image registration methods: A survey," *Image Vis. Comput.*, vol. 21, no. 11, pp. 977–1000, Oct. 2003.
[59] G. Turk and J. F. O'Brien, "Shape transformation using variational implicit functions," in *Proc. ACM SIGGRAPH Courses (SIGGRAPH)*, 2005, pp. 335–342.
[60] G. Turk and J. F. O'Brien, "Modelling with implicit surfaces that interpolate," *ACM Trans. Graph.*, vol. 21, no. 4, pp. 855–873, Oct. 2002.
[61] H.-K. Zhao, S. Osher, and R. Fedkiw, "Fast surface reconstruction using the level set method," in *Proc. IEEE Workshop Variational Level Set Methods Comput. Vis.*, Jul. 2001, pp. 194–201.
[62] S. F. Frisken, R. N. Perry, A. P. Rockwood, and T. R. Jones, "Adaptively sampled distance fields: A general representation of shape for computer graphics," in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 2000, pp. 249–254.
[63] K. Zhang, L. Zhang, H. Song, and D. Zhang, "Reinitialization-free level set evolution via reaction diffusion," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 258–271, Jan. 2013.
[64] D. Hartmann, M. Meinke, and W. Schröder, "The constrained reinitialization equation for level set methods," *J. Comput. Phys.*, vol. 229, no. 5, pp. 1514–1535, Mar. 2010.
[65] D. Hartmann, M. Meinke, and W. Schröder, "Differential equation based constrained reinitialization for level set methods," *J. Comput. Phys.*, vol. 227, no. 14, pp. 6821–6845, Jul. 2008.
[66] C. Li, C. Xu, C. Gui, and M. D. Fox, "Level set evolution without re-initialization: A new variational formulation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 430–436.
[67] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang, "A PDE-based fast local level set method," *J. Comput. Phys.*, vol. 155, no. 2, pp. 410–438, Nov. 1999.
[68] C. Zach, T. Pock, and H. Bischof, "A globally optimal algorithm for robust TV-L1 range image integration," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.
[69] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," *ACM SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 163–169, Aug. 1987.
[70] F. Huguet and F. Devernay, "A variational method for scene flow estimation from stereo sequences," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–7.
[71] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremers, "Efficient dense scene flow from sparse or dense stereo data," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2008, pp. 739–751.
[72] C. Vogel, K. Schindler, and S. Roth, "Piecewise rigid scene flow," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1377–1384.
[73] J. Quiroga, T. Brox, F. Devernay, and J. Crowley, "Dense semi-rigid scene flow estimation from RGBD images," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2014, pp. 567–582.
[74] M. Jaimez, C. Kerl, J. Gonzalez-Jimenez, and D. Cremers, "Fast odometry and scene flow from RGB-D cameras based on geometric clustering," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2017, pp. 3992–3999.
[75] M. Slavcheva, M. Baust, and S. Ilic, "Variational level set evolution for non-rigid 3D reconstruction from a single depth camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Feb. 24, 2020, doi: 10.1109/TPAMI.2020.2976065.
[76] A. Tsoli and A. A. Argyros, "Tracking deformable surfaces that undergo topological changes using an RGB-D camera," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 333–341.
[77] J. Solomon, M. Ben-Chen, A. Butscher, and L. Guibas, "As-killing-as-possible vector fields for planar deformation," *Comput. Graph. Forum*, vol. 30, no. 5, pp. 1543–1552, Aug. 2011.
[78] J. Calder, A. Mansouri, and A. Yezzi, "Image sharpening via Sobolev gradient flows," *SIAM J. Imag. Sci.*, vol. 3, no. 4, pp. 981–1014, Jan. 2010.
[79] R. W. Sumner, J. Schmid, and M. Pauly, "Embedded deformation for shape manipulation," *ACM Trans. Graph.*, vol. 26, no. 3, p. 80, Jul. 2007.
[80] N. Aspert, D. Santa-Cruz, and T. Ebrahimi, "MESH: Measuring errors between surfaces using the Hausdorff distance," in *Proc. IEEE Int. Conf. Multimedia Expo*, Aug. 2002, pp. 705–708.
[81] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.
[82] O. Kahler, V. A. Prisacariu, C. Y. Ren, X. Sun, P. Torr, and D. Murray, "Very high frame rate volumetric integration of depth images on mobile devices," *IEEE Trans. Vis. Comput. Graphics*, vol. 21, no. 11, pp. 1241–1250, Nov. 2015.
[83] J. H. Lee, H. Ha, Y. Dong, X. Tong, and M. H. Kim, "TextureFusion: High-quality texture acquisition for real-time RGB-D scanning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1272–1280.

Jiwoo Kang was born in South Korea, in 1987. He received the B.S. degree in electrical and electronic engineering from Yonsei University, Seoul, South Korea, in 2011, and the M.S. and Ph.D. degrees in electrical and electronic engineering from Yonsei University in 2019, through the integrated Ph.D. program. He worked as a Researcher at University-Industry Foundation in Yonsei University from September 2019 to November 2020, and as a Research Professor at BK21 Y-BASE R&E Institute from December 2020 to February 2022. He is currently working as an Assistant Professor at Sookmyung Women's University, Seoul. His research interests include computer graphics, computer vision, and image analysis.

Seongmin Lee was born in South Korea, in 1992. He received the B.S. degree in electronic and electrical engineering from Hongik University, Seoul, South Korea, in 2018. He is currently pursuing the M.S. and Ph.D. degrees with the Multidimensional Insight Laboratory, Yonsei University, Seoul. His research interests are in the areas of computer vision, computer graphics, and deep learning.

Mingyu Jang was born in South Korea, in 1994. He received the B.S. degree in electronic and electrical engineering from Gangneung-Wonju National University, Gangneung, South Korea, in 2019. He is currently pursuing the M.S. and Ph.D. degrees with the Multidimensional Insight Laboratory, Yonsei University, Seoul. His research interests are in the areas of computer vision, computer graphics, and deep learning.

Sanghoon Lee (Senior Member, IEEE) received the B.S. degree from Yonsei University, Seoul, South Korea, in 1989, the M.S. degree from the Korea Advanced Institute of Science and Technology, Daejeon, South Korea, in 1991, and the Ph.D. degree from The University of Texas at Austin, Austin, TX, USA, in 2000. From 1991 to 1996, he was with Korea Telecom, Cheongju, South Korea. From 1999 to 2002, he was with Lucent Technologies, Murray Hill, NJ, USA. In 2003, he was a Faculty Member with the Department of Electrical Engineering, Yonsei University, where he is currently a Full Professor. His current research interests include image/video processing, computer vision, and graphics. From 2014 to 2019, he was a member of the IEEE Image, Video, and Multidimensional Signal Processing Workshop (IVMSP) Technical Committee. Since 2016, he has been a member of the IEEE International Workshop on Multimedia Signal Processing (MMSP) Technical Committee. He served as a steering committee member for the IEEE and APISPA conferences. He was a Board of Governors Member of the Asia-Pacific Signal and Information Processing Association (APSIPA) in 2020. He received the 2012 Special Service Award from the IEEE Broadcast Technology Society, the 2013 Special Service Award from the IEEE Signal Processing Society, and the 2015 Yonsei Academic Award from Yonsei University. He was the General Chair of the 2013 IEEE IVMSP Workshop. He has been serving as the Chair for the IEEE P3333.1 Quality Assessment Working Group since 2011. From 2018 to 2019, he was the Image, Video, and Multimedia (IVM) Technical Committee Chair of APSIPA. He served as an Editor for the *Journal of Communications and Networks* from 2009 to 2015 and a Special Issue Guest Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING in 2013. From 2010 to 2014, he was an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING. From 2014 to 2018, he served as an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS. He is currently a Senior Area Editor of the IEEE SIGNAL PROCESSING LETTERS.