

Thin-Shell-SfT: Fine-Grained Monocular Non-rigid 3D Surface Tracking with Neural Deformation Fields

Supplementary Material

Table of Contents

A. Thin Shell Physical Prior	1
B. Variants of the Temporal Constraint	2
C. Additional Evaluations	2
D. Hyperparameters	2
E. Detailed Ablations	3
F. Qualitative Results	3
G. Limitations	3

and curvature tensors on the initial surface $\bar{\mathbf{x}}$. This basis includes $\bar{\mathbf{a}}_\alpha$, the set of two vectors tangential to the curvilinear coordinate lines ξ^α :

$$\bar{\mathbf{a}}_\alpha := \bar{\mathbf{x}}_{,\alpha}. \quad (9)$$

The local unit normal $\bar{\mathbf{a}}_3$, is then computed as the cross product of the tangent base vectors:

$$\bar{\mathbf{a}}_3 := \frac{\bar{\mathbf{a}}_1 \times \bar{\mathbf{a}}_2}{|\bar{\mathbf{a}}_1 \times \bar{\mathbf{a}}_2|}, \quad \bar{\mathbf{a}}^3 = \bar{\mathbf{a}}_3. \quad (10)$$

The local basis $\{\bar{\mathbf{a}}_1, \bar{\mathbf{a}}_2, \bar{\mathbf{a}}_3\}$ is additionally used as per-point rotation matrix \mathbf{R} for the Gaussian tracking (see Sec. 4.2).

The surface area differential $d\Omega$ relates to the curvilinear coordinates via the Jacobian of the metric tensor:

$$d\Omega = \sqrt{a} d\xi^1 d\xi^2, \quad \text{where } \sqrt{a} := |\bar{\mathbf{a}}_1 \times \bar{\mathbf{a}}_2|. \quad (11)$$

A. Thin Shell Physical Prior

This section provides additional details on the physics prior, focusing on the differential geometric computations on the parameterised initial and deformed surfaces. First, we write down the detailed notations used in the main matter and this appendix. Next, we describe the geometric quantities and computations required for evaluating strain in Eq. (6)-(main matter), and subsequently the physics loss (Eq. (8)-(main matter)).

Notations. Following NeuralClothSim [34], we use Greek letters for indexing quantities on the 2D-dimensional surface (e.g., $\mathbf{a}_\alpha, \alpha, \beta, \dots = 1, 2$). An index can appear as a superscript or subscript. Superscripts $(\cdot)^\alpha$ refer to contravariant tensor components, which scale inversely with the change of basis; subscripts $(\cdot)_\alpha$ refer to covariant components that change in the same way as the basis scale. We use upper dot notation for time derivatives; vertical bar for covariant derivatives; and lower comma notation for partial derivatives w.r.t. the curvilinear coordinates, ξ^α (e.g., $\dot{\mathbf{u}} = \partial \mathbf{u} / \partial t, u_{\lambda|\alpha}$, and $\mathbf{u}_{,\alpha} = \partial \mathbf{u} / \partial \xi^\alpha$, respectively). Moreover, geometric quantities with overbar notation $(\bar{\cdot})$ refer to the initial surface state, and Einstein summation convention of repeated indices is used for tensorial operations (e.g., $\varphi_{\alpha\lambda} \varphi_\beta^\lambda = \varphi_{\alpha 1} \varphi_\beta^1 + \varphi_{\alpha 2} \varphi_\beta^2$). For notational clarity, we drop the input ξ, t and parameters Υ, Θ in all the derived quantities (e.g., $\bar{\mathbf{a}}_1(\xi; \Upsilon), \varepsilon(\xi, t; \Upsilon, \Theta)$).

Covariant Basis. In the first step, we define a local covariant basis to express local quantities such as the metric

Metric Tensor and Contravariant Basis. The covariant components of the symmetric metric tensor (*i.e.*, first fundamental form) that measures the distortion of length and angles are computed as:

$$\bar{a}_{\alpha\beta} = \bar{a}_{\beta\alpha} := \bar{\mathbf{a}}_\alpha \cdot \bar{\mathbf{a}}_\beta. \quad (12)$$

The corresponding contravariant components of the symmetric metric tensor denoted by $\bar{a}^{\alpha\lambda}$ are obtained using the identity: $\bar{a}^{\alpha\lambda} \bar{a}_{\lambda\beta} = \delta_{\alpha\beta}$, where $\delta_{\alpha\beta}$ stands for the Kronecker delta. $\bar{a}^{\alpha\lambda}$ can be used to compute the contravariant basis vectors as follows: $\bar{\mathbf{a}}^\alpha = \bar{a}^{\alpha\lambda} \bar{\mathbf{a}}_\lambda$. While the covariant base vector $\bar{\mathbf{a}}_\alpha$ is tangent to the ξ^α line, the contravariant base vector $\bar{\mathbf{a}}^\alpha$ is normal to $\bar{\mathbf{a}}_\beta$ when $\alpha \neq \beta$. Note that $\bar{\mathbf{a}}_\alpha$ and $\bar{\mathbf{a}}^\alpha$ are not necessarily unit vectors.

Curvature Tensor. The curvature metric of the initial surface (*i.e.* the second fundamental form) is computed as follows:

$$\bar{b}_{\alpha\beta} := -\bar{\mathbf{a}}_\alpha \cdot \bar{\mathbf{a}}_{3,\beta} = -\bar{\mathbf{a}}_\beta \cdot \bar{\mathbf{a}}_{3,\alpha} = \bar{\mathbf{a}}_{\alpha,\beta} \cdot \bar{\mathbf{a}}_3. \quad (13)$$

Covariant Derivatives. When taking derivatives along a curve on the midsurface, we must account for the change of the local basis along that curve. More concretely, we rely on the *surface covariant derivative* to evaluate the deformation gradient $\mathbf{u}_{,\alpha}$ on the deformed midsurface in Eq. (5) of the main paper. We compute the covariant derivatives of the *deformed surface* quantities, *i.e.* first-order tensor $u_\lambda|_\alpha$

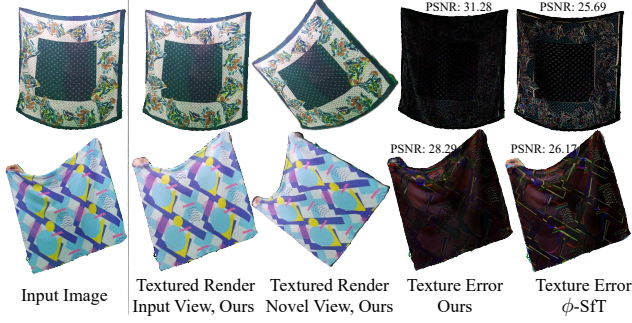


Figure I. **Dynamic novel-view synthesis.** We render the tracked Gaussians from input and novel views and visualise the texture error (ℓ_1 loss) for the input view. Our lower texture error compared to previous SotA enables higher-fidelity surface reconstructions.

Metric	Method	S1	S2	S3	S4	S5	S6	S7	S8	S9	Avg
NC-Cos \downarrow	ϕ -SfT	0.022	0.064	0.059	0.037	0.044	0.028	0.058	0.030	0.026	0.041
	Ours	0.029	0.015	0.071	0.036	0.062	0.014	0.039	0.017	0.022	0.034
NC- $\ell_2 \downarrow$	ϕ -SfT	0.006	0.011	0.014	0.013	0.016	0.013	0.016	0.013	0.012	0.013
	Ours	0.005	0.006	0.016	0.009	0.015	0.007	0.010	0.008	0.008	0.009
PSNR \uparrow	ϕ -SfT	28.17	26.82	26.17	27.49	25.18	25.08	23.15	25.69	26.39	26.02
	Ours	32.83	31.93	28.29	30.81	30.22	31.24	29.09	31.28	31.34	30.78
LPIPS \downarrow	ϕ -SfT	0.021	0.027	0.040	0.045	0.055	0.048	0.049	0.049	0.045	0.042
	Ours	0.013	0.021	0.052	0.049	0.043	0.066	0.054	0.042	0.043	0.042
Runtime \downarrow	ϕ -SfT	20h3m	6h23m	15h45m	21h33m	9h10m	5h25m	9h40m	17h1m	18h3m	13h40m
	Ours	47m	28m	38m	36m	43m	34m	36m	37m	39m	38m

Table I. **Additional metrics and comparisons.** We compare with ϕ -SfT on image-based metrics, including cosine and ℓ_2 normal consistency error; our Thin-Shell-SfT generates more accurate results while being significantly faster than ϕ -SfT.

and the second-order tensor $\varphi_{\alpha\lambda|\beta}$ in Eqs. (5) and (6)-(main matter), using the following rules:

$$u_{\alpha|\beta} = u_{\alpha,\beta} - u_{\lambda}\Gamma_{\alpha\beta}^{\lambda}, \text{ and} \quad (14)$$

$$\varphi_{\alpha\beta|\gamma} = \varphi_{\alpha\beta,\gamma} - \varphi_{\lambda\beta}\Gamma_{\alpha\gamma}^{\lambda} - \varphi_{\alpha\lambda}\Gamma_{\beta\gamma}^{\lambda},$$

where the Christoffel symbol $\Gamma_{\alpha\beta}^{\lambda}$ is defined as (similarly for $\Gamma_{\alpha\gamma}^{\lambda}$ and $\Gamma_{\beta\gamma}^{\lambda}$),

$$\Gamma_{\alpha\beta}^{\lambda} := \bar{\mathbf{a}}^{\lambda} \cdot \bar{\mathbf{a}}_{\alpha,\beta}. \quad (15)$$

Symmetric Tensors. We exploit the symmetry with respect to indices α and β , *i.e.* $a_{\alpha\beta} = a_{\beta\alpha}$, for efficient computations of the following tensors: $\bar{a}_{\alpha\beta}$, $\bar{b}_{\alpha\beta}$, $\varepsilon_{\alpha\beta}$, $\kappa_{\alpha\beta}$, and $\Gamma_{\alpha\beta}^{\lambda}$. The fourth-order symmetric tensor \mathbf{H} , as in Eq. (7)-(main matter), uses:

$$H^{\alpha\beta\lambda\delta} = H^{\beta\alpha\lambda\delta} = H^{\beta\alpha\delta\lambda} = H^{\alpha\beta\delta\lambda} = H^{\lambda\delta\alpha\beta}.$$

This property means that only six independent components (after applying symmetry) need to be computed (*i.e.*, H^{1111} , H^{1112} , H^{1122} , H^{1212} , H^{1222} , and H^{2222}).

B. Variants of the Temporal Constraint

We proposed a momentum regulariser in our deformation formulation (Eq. (2)-(main matter)). Along with this, as

mentioned in the main matter, we experimented with two other variants of temporal consistency that gave improved qualitative results for two of the nine ϕ -SfT sequences (S3 and S4). For these variants, we reformulated the deformed point position on the tracked surface as

$$\mathbf{x}(\xi, t) = \bar{\mathbf{x}}(\xi) + \mathbf{u}(\xi, t) \quad (16)$$

with $\mathbf{u}(\xi, t) = \mathcal{F}(\xi, t), \forall t \in [1, \dots, T],$

where we directly regress the deformation (NDF) as the offset to the initial state using MLP $\mathcal{F}(\cdot)$. As $\mathbf{u}(\xi, 1) = 0$ is no longer implicit (unlike Eq. (2)), the total loss \mathcal{L} now additionally includes minimisation objectives of (a) initial deformation $\mathbf{u}(\xi, 1; \Theta)$ and (b) either acceleration $\ddot{\mathbf{u}}(\xi, t; \Theta)$ (variant I, S3) or velocity $\dot{\mathbf{u}}(\xi, t; \Theta)$ (variant II, S4).

Regarding λ . For the momentum regulariser (Eq. (2)-(main matter)), we tried $\lambda=1$ instead of the proposed value $\lambda=0.4$ in our early experiments. In that case, the network prediction $F(\xi, t)$ would have an alternate interpretation of velocity instead of deformation offset. However, this led to noisier initialisation of the later surface states due to accumulated offset and, hence, noisy optimisation; thus, we decided upon a $\lambda < 1$. Note that λ is positive to encourage deformation follow-through (more details in Appendix E).

C. Additional Evaluations

Dynamic Novel View Synthesis. Although our work focuses on deformable surface tracking but not directly novel view synthesis or appearance reconstruction, we additionally show textured tracking and compute the PSNR and LPIPS from input views (ground truth is not available for novel views); see Fig. I and Tab. I.

Normal Maps. In addition to the Chamfer distance (Tab. 1), we evaluate our reconstructions with another image-based metric, *i.e.* cosine and ℓ_2 normal consistency (following Refs. [28, 68]); see Tab. I and Fig. 4-(main matter) for all results. The normal metric captures the error in the fine-grained details of the reconstructions, where we notably outperform the previous SotA (ϕ -SfT).

Runtime. In Tab. I, we report the runtime for each sequence. Thin-Shell-SfT typically takes between 30 minutes and one hour until convergence on an NVIDIA A100 GPU. Although computationally expensive, ours is significantly faster ($\approx 38\times$) than ϕ -SfT [33], which takes up to 16–24 hours. While a recent method [67] takes up to three minutes, our method significantly outperforms both in the fine-grained wrinkle reconstruction.

D. Hyperparameters

Number of Gaussians. In Fig. III, we report the reconstruction error, visualise the surfaces for varying Gaussian counts, and observe the reconstruction quality drops only

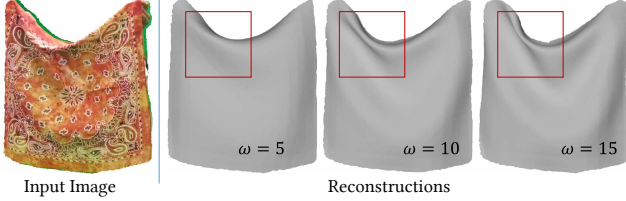


Figure II. **Sharpness control.** The amount of deformation details in the reconstructions can be tuned by varying NDF ω .

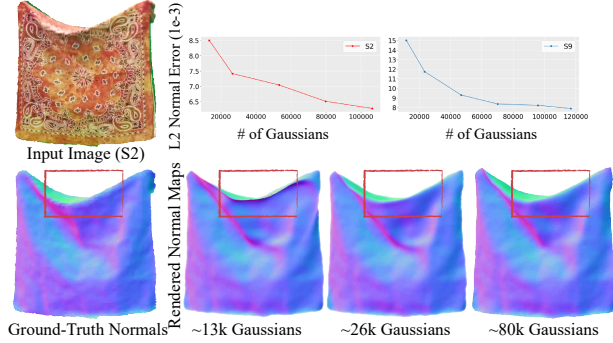


Figure III. **Gaussian count.** (top:) Reconstruction error for the varying number of Gaussians (N_g); (bottom:) Even at lower N_g s, our method tracks surfaces with fine-grained details, although with slightly lesser accuracy.

Seq	S1	S2	S3	S4	S5	S6	S7	S8	S9	Avg
w/o physics prior	1.96	1.65	7.01	35.28	56.60	15.29	117.5	41.36	31.61	34.25
w/o surface Gaussians	5.96	20.77	9.46	49.72	13.71	8.80	6.80	5.40	6.41	14.00
w/o momentum	1.52	1.00	3.67	6.02	8.94	3.15	4.87	2.25	3.56	3.89
w/o normal scale	1.20	0.53	3.14	5.40	8.69	3.07	5.14	2.30	4.29	3.75
w/o mask loss	1.61	0.62	3.49	5.67	9.05	2.91	4.24	1.90	2.26	3.53
Ours (full)	1.17	0.55	3.49	5.66	8.69	2.51	3.80	2.27	3.00	3.46

Table II. **Detailed ablations.** We report the Chamfer distance for all sequences of ϕ -SfT dataset when ablating various design choices: without the physics loss, without surface-induced Gaussian parameters, without fixing Gaussian scale along the surface normal, without momentum regularisation, and without mask loss.

slightly with fewer Gaussians. The number of Gaussian samples in our main experiments is $N_g \approx 80\text{--}90k$.

Smoothness Control. Depending on the application, it could be useful to control the smoothness and sharpness of the reconstructed surface. This can be achieved by tuning the frequency ω (set to default 30 in all our experiments) of sinusoidal activation [66] in the NDF; see Fig. II.

E. Detailed Ablations

We provided a detailed description (Sec. 5.2), the qualitative results (Fig. 7) and summarised ablation results in the main matter. Here, we additionally report the results on the full ϕ -SfT dataset and provide three additional ablations. We test the following modes: 1) Without Kirchhoff-Love thin-shell-based physical prior, 2) No surface-induced

Gaussians, *i.e.*, optimising Gaussian parameters (*i.e.*, scale, opacity, and colour) on all input frames instead of the single (template) frame, and 3) Without fixing the scale along the surface normal, 4) Without the momentum regularisation, and 5) Without mask loss. In Tab. II, we report the error to the ground truth for all the ablated versions on each sequence. We notice that including *continuous* physics loss and surface-induced Gaussians are crucial for accurate surface tracking.

No Fixed Normal Scale. Regarding the surface-induced Gaussians, we test the variant that optimises the 3D scales $(s_1, s_2, s_3)_i$ and rotation \mathbf{R}_i of each Gaussian in \mathcal{G}_1 instead of setting $s_3 := \epsilon$ and $\mathbf{R}_i = [\bar{\mathbf{a}}_1 \ \bar{\mathbf{a}}_2 \ \bar{\mathbf{a}}_3]_i$, as in our full method. 1) Missing normal scale regularisation leads to elongated 3D Gaussians along the view direction, leading to a high RGB loss; see Fig. 7-(d)-(main matter).

Momentum Regularisation. We perform joint space-time NDF optimisation while enforcing backpropagation of information to previous states using (Eq. (2)-(main matter)). By setting $\lambda = 0$, we test the variant with no explicit temporal constraint. This reduces the accuracy as reported in Tab. II. The momentum term encourages the current deformed state to follow the previous deformation. It especially helps in sequences with large sway (*e.g.*, single-wrinkled S2) and is less effective for frequently alternating deformations.

Mask Loss. Masks are optional inputs for our method. When using mask loss, we observe a speedup in convergence ($1.5\times$ faster) but did not notice much qualitative or quantitative improvement in surface tracking.

F. Qualitative Results

Our Thin-Shell-SfT outperforms the existing methods, especially qualitatively and for *fine-grained* details such as wrinkles. We visualise reconstructions of our Thin-Shell-SfT on two ϕ -SfT sequences; see Fig. IV. The figure shows the input image sequences of the evolving surface and their corresponding spatiotemporally coherent 3D reconstructions for selected frames. Please refer to the supplemental video for the visualisation of surface tracking of all sequences.

G. Limitations

Our method reconstructs the challenging fine-grained surface deformations from monocular videos. Thanks to the physics prior, the method is reasonably robust to occlusions although we notice self-collision in extreme cases, as this is not explicitly handled; see Fig. V. Since the surface can cast self-shadows, non-Lambertian surfaces can appear differently over time. While our approach remains robust against changes in appearance across frames for the tested dataset, substantial changes (*e.g.* specular surfaces) can lead to a de-

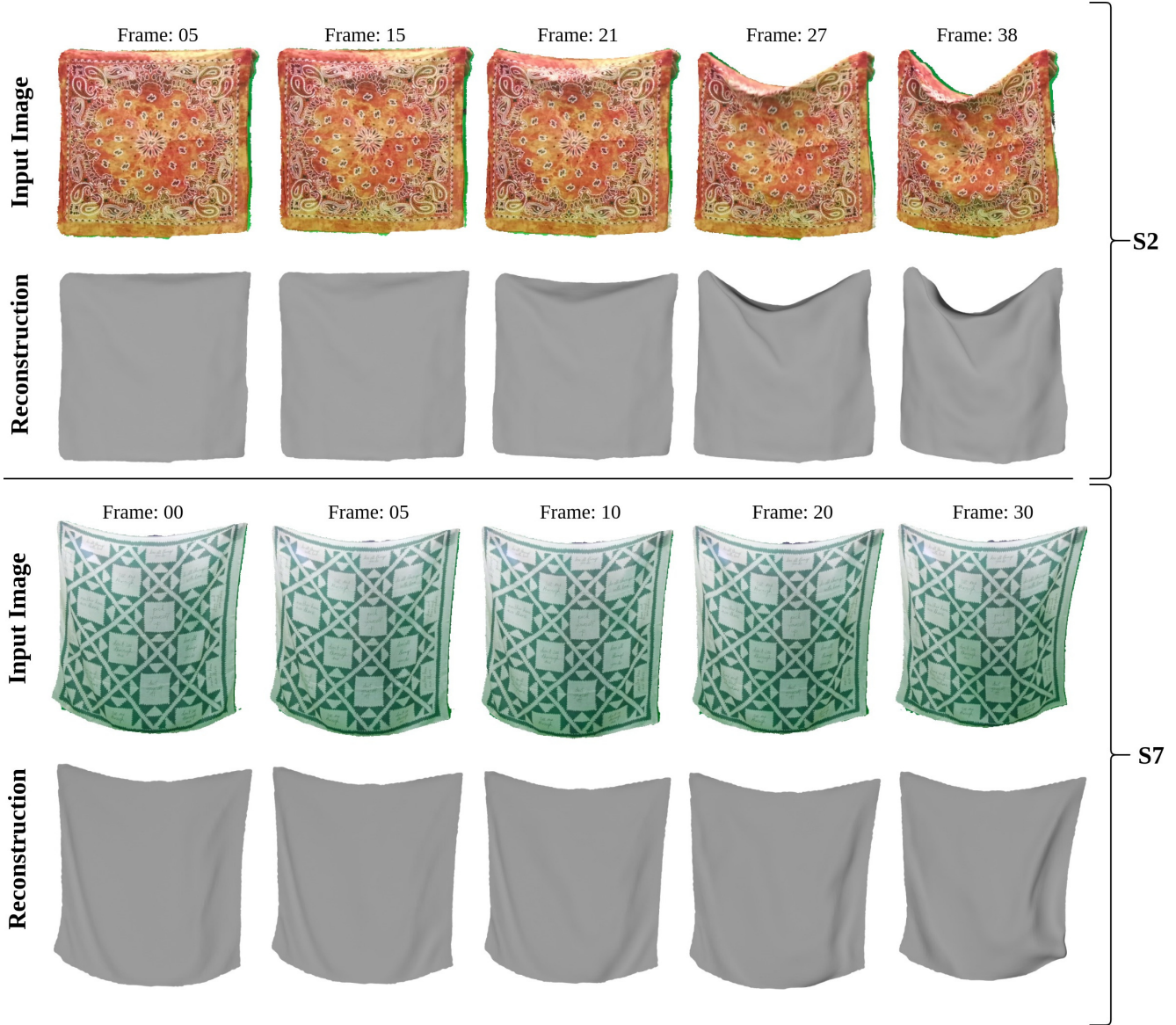


Figure IV. **Exemplary 4D surface tracking results by Thin-Shell-SfT.** We show additional qualitative results on two sequences. Our method can reconstruct high-quality wrinkles and deformations just from the monocular video. Please see the supplementary video for tracked reconstructions of all sequences.

cline in 3D reconstruction quality. Similarly, tracking of textureless surfaces is yet another important problem; we leave it as future work. Overall, we significantly improve surface tracking using an *adaptive* deformation model, *continuous* thin shell loss and surface-induced 3D Gaussian Splatting compared to existing approaches.

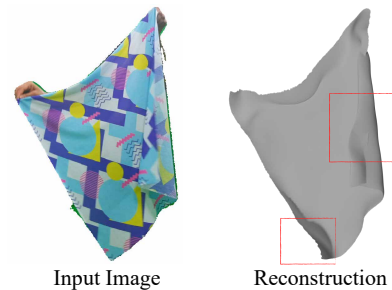


Figure V. **Visualisation of the limitation.** Our method does not handle the self-collision of tracked surfaces. Moreover, appearance changes due to deformation (*e.g.*, shadows) can lead to minor artefacts.