

Supplementary Materials: IF-Garments: Reconstructing Your Intersection-Free Multi-Layered Garments from Monocular Videos

Anonymous Authors

In this supplementary material, we will further detail the following aspects omitted in the main paper:

- In Appendix A, we describe the detailed algorithm of the solver of our physics-aware module to promote the understanding of the simulation method.
- In Appendix B, we show the improvement of post-processing, including isolated elements removing, filling holes, and Laplacian smoothing.
- In Appendix C, we conduct an ablation study on the masked Signed Distance Field (SDF) to illustrate our comprehensive consideration of handling intersection.
- In Appendix D, we demonstrate a potential application of our work about clothing matching.
- In Appendix E, we present the network architecture of the canonical SDF ϕ and non-rigid deformation field d .

A THE SOLVER OF PHYSICS-AWARE MODULE

As mentioned in Section 3.3 of the main paper, the physics-aware module employs a position-based dynamics framework[1, 3, 5], which is popular in soft body simulation[2, 6].

We demonstrate that the algorithm described in Section 3.3.3 of the main paper is responsible for establishing constraints. Here, we explain the solving details, as illustrated in Algorithm 1. From line 1 to line 3, Δt donates the time step size, h for substep size, \mathbf{p} for position, and \mathbf{v} for velocity. Constraint compliance is denoted by β (line 5), representing the strength of the constraint[3]. All of the constraints are applied to the position of vertices. The collision constraint C^{coll} is solved immediately (line 6 to line 10) as well as the air constraint C^{air} (line 11 to line 15), while the stretch constraint C^{str} is iteratively satisfied through the manifold projection (line 16 to line 22). line 13 is related to Equation (7) in the main paper, where $\hat{\mathbf{p}}$ and $\hat{\mathbf{q}}$ are the nearest points pair belonging to two adjacent layers and computed at the beginning of the simulation serving as the air gap vector. At the end of the substep, velocities are updated (line 24) based on the final positions (line 23). For each garment layer, the time step size Δt is set to 0.01s, and the number of time steps and sub-steps to 360 and 10, respectively. For C^{str} , we have $\beta = 1e^{-5}$.

B POST-PROCESSING

As introduced at the end of Section 3.2, we additionally conduct post-processing in single garment extraction to mitigate the noise caused by segmentation errors and self-occlusion in self-rotating videos, including isolated elements removal (see Figure 1), hole filling (see Figure 2(a)), and Laplacian smoothing (see Figure 2(b)). Please note that not all garments suffer from isolated elements and holes, thus we only present some typical examples. Laplacian smoothing is imposed on the boundary of all garments.

Algorithm 1 The solver of physics-aware module.

```

1:  $h \leftarrow \Delta t / \text{numSubSteps}$  ▷ Substep size
2: for  $i = 1, \dots, \text{numSubSteps}$  do ▷ Iteration
3:   initialize solve  $\mathbf{p} \leftarrow \mathbf{p}^i + h\mathbf{v}^i$ 
4:   initialize multipliers  $\lambda \leftarrow 0$ 
5:    $\tilde{\beta} \leftarrow \frac{1}{h^2}\beta$ 
6:   for all collision constraint  $C^{coll}$  do
7:     if  $C^{coll}(\mathbf{p}) < 0$  then
8:        $\mathbf{p} \leftarrow \mathbf{p} - \nabla\phi(\mathbf{p})\phi(\mathbf{p})$ 
9:     end if
10:  end for
11:  for all air constraint  $C^{air}$  do
12:    if  $C^{air}(\mathbf{p}) < 0$  then
13:       $\mathbf{p} \leftarrow \mathbf{p} - (\hat{\mathbf{p}} - \hat{\mathbf{q}})$ 
14:    end if
15:  end for
16:  for all stretch constraint  $C^{str}$  do
17:     $\mathbf{A} \leftarrow \nabla C(\mathbf{x})\mathbf{M}^{-1}\nabla C^T(\mathbf{p}) + \tilde{\beta}$ 
18:     $\Delta\lambda \leftarrow -\mathbf{A}^{-1}(C(\mathbf{p}) + \beta\lambda)$ 
19:     $\Delta\mathbf{p} \leftarrow \mathbf{M}^{-1}\nabla C^T(\mathbf{p})\Delta\lambda$ 
20:     $\lambda \leftarrow \lambda + \Delta\lambda$ 
21:     $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$ 
22:  end for
23:  update positions  $\mathbf{p}^{i+1} \leftarrow \mathbf{p}$ 
24:  update velocities  $\mathbf{v}^{i+1} \leftarrow \frac{\mathbf{p}^{i+1} - \mathbf{p}^i}{h}$ 
25:   $i \leftarrow i + 1$ 
26: end for

```

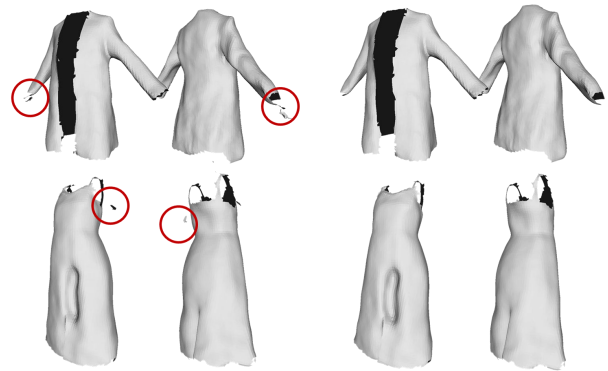


Figure 1: Two examples of removing isolated elements. Though segmentation errors lead to some isolated elements, it is easy to handle.

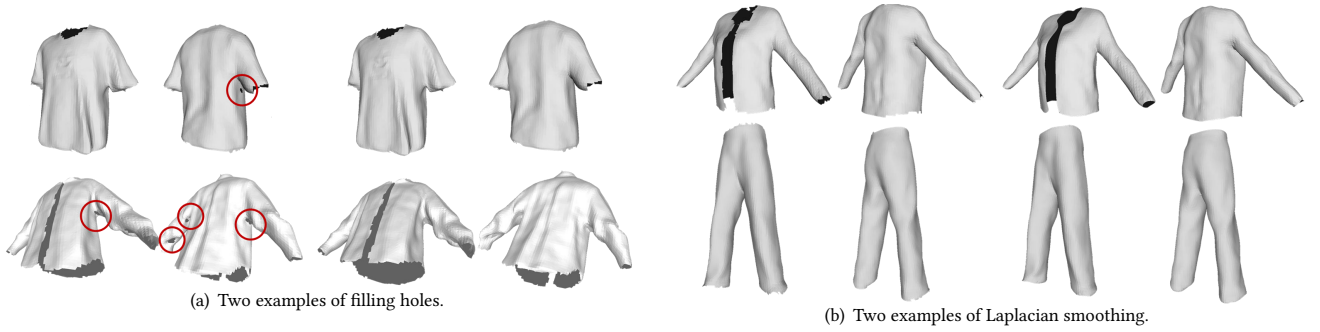


Figure 2: Examples of filling holes and Laplacian smoothing. (a) There are a few holes in areas prone to be occluded, such as the armpits. (b) We adopt Laplacian smoothing to refine the boundary of the garments.

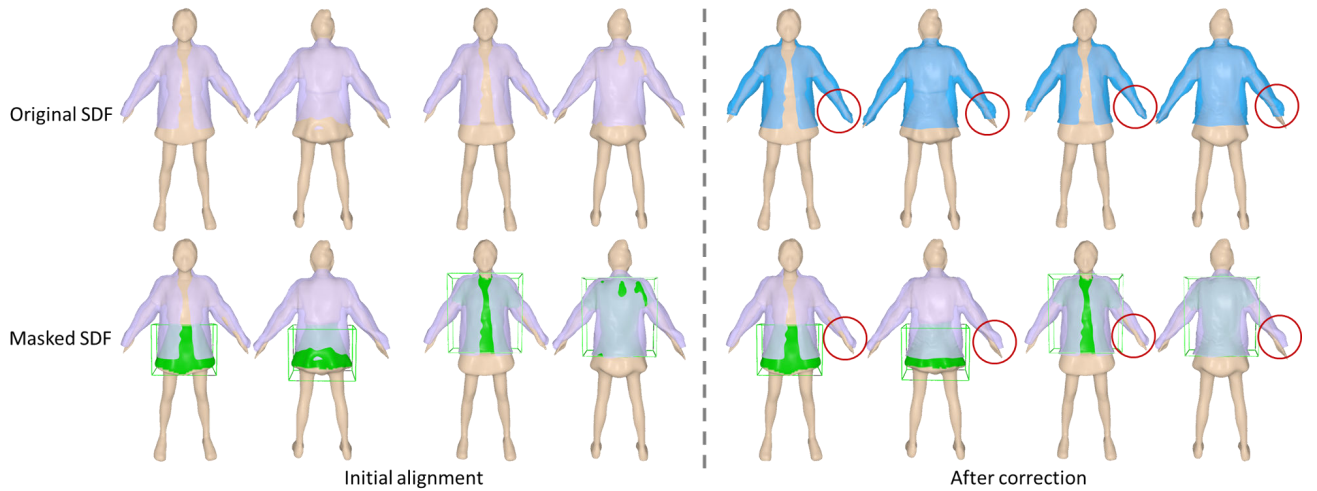


Figure 3: Comparison of correction with the original SDF and the masked SDF. The top row uses the original SDF while the bottom row uses the masked SDF. The mask is calculated by the bounding box of the garment (green). The blue coat (top right) is rectified by the original SDF and suffers from unnecessary deformations around the cuffs. With the masked SDF (bottom right), the intersections between the coat and both the short skirt and the T-shirt are removed while keeping other areas unaffected.

C THE EFFECT OF MASKED SDF

As described in Section 3.3.3 in the main paper, the SDF of the clothed body is leveraged to handle intersection. The goal is to eliminate the inter-penetration between garments rather than the whole clothed body. The original SDF will lead to unnecessary corrections due to irrelevant parts of the clothed body. Therefore, we implement a masked SDF computed based on the garment's Bounding BoX (BBX). The effect of the masked SDF is shown in Figure 3. Specifically, the jacket (purple) intersects with the short skirt and the T-shirt in the initial alignment. Thus the original SDFs (top row) of the short skirt and the T-shirt are employed to correct the jacket. The masks of the short skirt and the T-shirt are calculated by their BBX (bottom row), which means if a vertex of the jacket is not in the range of the BBX, its collision constraint will be deactivated. For correction based on the original SDF, the jacket exhibits significant unnecessary deformations around the cuffs (blue). In contrast, thanks to the masked SDF, we successfully

ensure that the collision constraint only affects the clothing area, rather than other parts of the clothed body.

D APPLICATION ON CLOTHING MATCHING

Recently, it has been popular to purchase clothes on e-commerce platforms. Though consumers can find clothing-matching recommendations on live streams or social media apps, they cannot freely try out combinations of garments they prefer. Consider an application where users including customers and models employed by the sellers upload self-rotating videos of people wearing clothing, accompanied by our well-generalizable method for clothing digitization to build a large-scale 3D asset library. Therefore, users are empowered to find garments and make a combination to preview the matching results, facilitated by our robust physics-aware module to ensure intersection-free. Figure 4 visually illustrates this interesting idea. We believe such a visionary application will help people find more satisfying outfits.

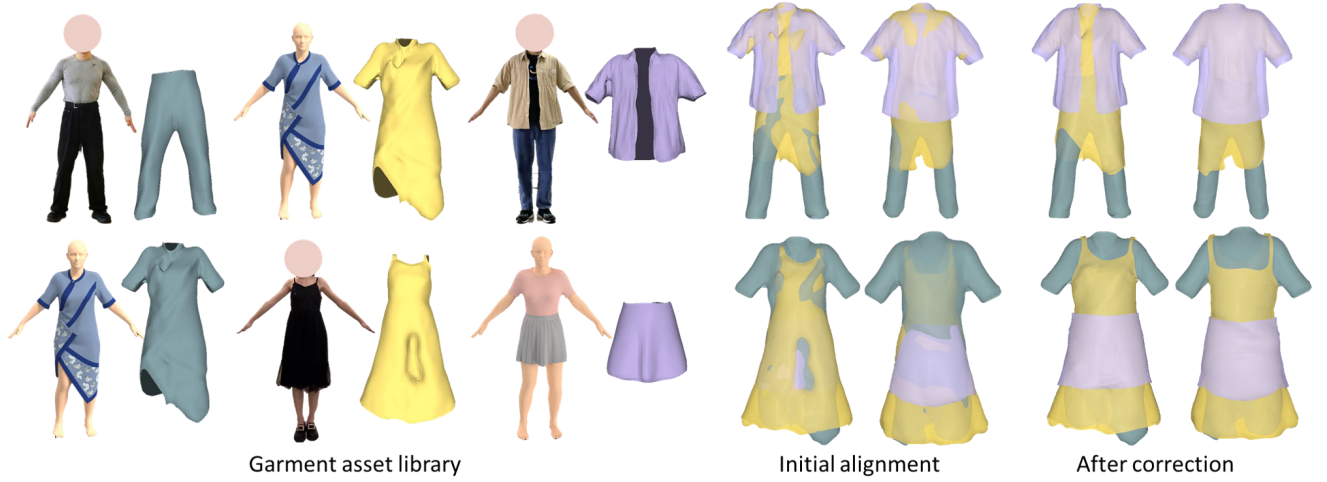


Figure 4: An application about clothing matching. Due to the robust capability of handling intersections, IF-Garments presents promise in clothing matching that enables the customer to combine various kinds of garments according to their preferences.

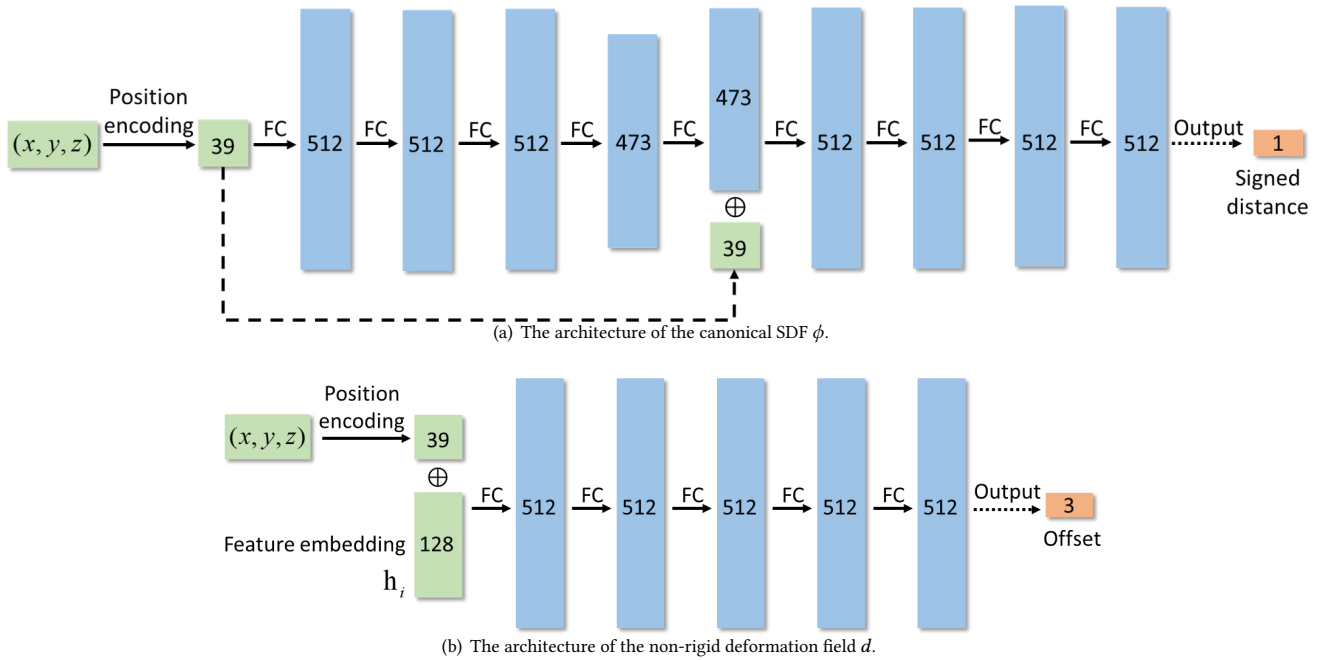


Figure 5: Details of MLPs adopted in our work. Rectangles represent vectors, arrows stand for operations, FC means the fully connected layers, and \oplus is concatenation. We leverage position encoding[4] to enhance the learning of the spatial features of the clothed body. In (b), d is conditioned by the feature embedding h_i of i -th frame.

E NEURAL NETWORK ARCHITECTURE

In Section 3.1 of the main paper, Multi-Layer Perceptrons (MLPs) are employed to represent the SDF ϕ in canonical space and the non-rigid deformation field d . Their architecture is shown in Figure 5.

REFERENCES

- [1] Jan Bender, Matthias Müller, and Miles Macklin. 2015. Position-Based Simulation Methods in Computer Graphics.. In *Eurographics (tutorials)*. 8.
- [2] Jerry Hsu, Tongtong Wang, Zherong Pan, Xifeng Gao, Cem Yuksel, and Kui Wu. 2023. Sag-Free Initialization for Strand-Based Hybrid Hair Simulation. *ACM Transactions on Graphics (TOG)* 42, 4 (2023), 1–14.
- [3] Miles Macklin, Matthias Müller, and Nuttapong Chentanez. 2016. XPBD: position-based simulation of compliant constrained dynamics. In *Proceedings of the 9th International Conference on Motion in Games*. 49–54.
- [4] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* 65, 1 (2021), 99–106.
- [5] Matthias Müller, Bruno Heidelberger, Marcus Hennix, and John Ratcliff. 2007. Position based dynamics. *Journal of Visual Communication and Image Representation* 18, 2 (2007), 109–118.
- [6] Tuur Stuyck and Hsiao-yu Chen. 2023. Diffxpbd: Differentiable position-based simulation of compliant constraint dynamics. *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 6, 3 (2023), 1–14.