Normal-guided Garment UV Prediction for Human Re-texturing

Yasamin Jafarian[†] Tuanfeng Y. Wang[♯] Duygu Ceylan[♯] Jimei Yang[♯]
Nathan Carr[♯] Yi Zhou[♯] Hyun Soo Park[†]

[†]University of Minnesota [♯]Adobe Research

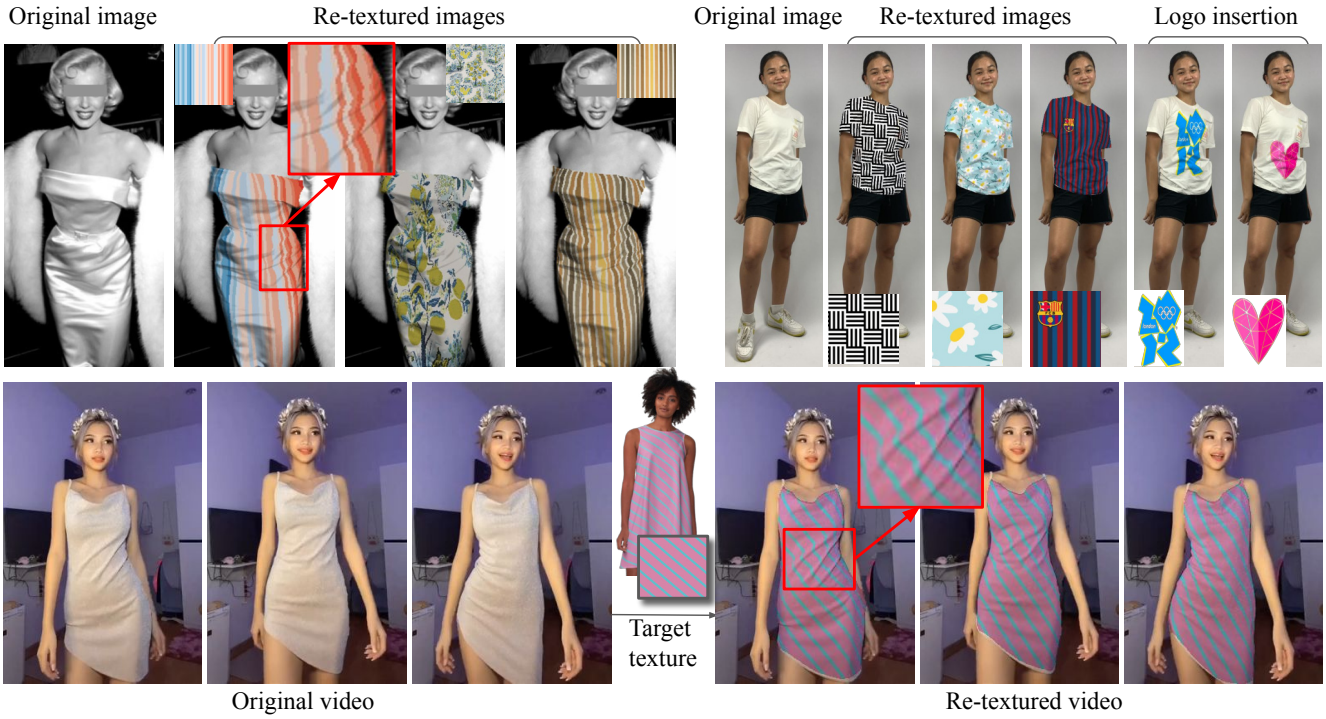

Figure 1. **Geomety aware texture editing from images and videos.** This paper presents a novel approach to predict a geometry aware texture (UV) map of a garment from an image (first row) and video (second row). The predicted UV map preserves isometry between texture space and 3D surface by leveraging 3D surface normals. Further, we ensure temporal consistency in the predicted UV map across frames in a video, resulting in physically plausible human appearance editing. Project website: yasamin.page/normal-guided-uv.

Abstract

Clothes undergo complex geometric deformations, which lead to appearance changes. To edit human videos in a physically plausible way, a texture map must take into account not only the garment transformation induced by the body movements and clothes fitting, but also its 3D fine-grained surface geometry. This poses, however, a new challenge of 3D reconstruction of dynamic clothes from an image or a video. In this paper, we show that it is possible to edit dressed human images and videos without 3D reconstruction. We estimate a geometry aware texture map between the garment region in an image and the texture space, a.k.a, UV map. Our UV map is designed to preserve isometry with respect to the underlying 3D surface by making use of the 3D surface normals predicted from the image. Our approach captures the underlying geometry of the garment in a self-supervised way, requiring no ground truth annotation of UV maps and can be readily extended to predict temporally coherent UV maps. We demonstrate that our method outperforms the state-of-the-art human UV map estimation approaches on both real and synthetic data.

1. Introduction

While browsing online clothing shops, have you ever wondered how the appearance of a dress of interest would look on you as if you were in a fitting room given your dress with a similar shape? A key technology to enable generating such visual experiences is *photorealistic re-texturing*—editing the texture of clothes in response to the subject's movement in the presented images or videos in a geomet-

rically and temporally coherent way. Over the past few years, there has been a significant advancement in the image and video editing technologies [4–6, 11–13, 17, 20, 21, 26, 31–33, 38, 46, 54], such as inserting advertising logos on videos of moving cars or applying face makeup on social media. However, such editing approaches designed for rigid or semi-rigid surfaces are not suitable for garments that undergo complex secondary motion with respect to the underlying body. For example, the fine wrinkles of the dress in Figure 1 result in complex warps in texture over time. In this paper, we present a new method to edit the appearance of a garment in a given image or video by taking into account its fine-grained geometric deformation.

Previous works address photorealistic texture editing in two ways. (1) 3D reconstruction and rendering: these approaches can achieve high-fidelity texture editing given highly accurate 3D geometry. On other side of the coin, their performance is dictated by the quality of the 3D reconstruction. While the 3D geometry of the garment can be learned from paired human appearance data, e.g., human modeling repositories with 3D meshes and renderings [1], due to the scarcity of such data, it often cannot generalize well on unseen real images and videos. (2) Direct texture mapping: by estimating dense UV map, these methods can bypass the procedure of 3D reconstruction [18, 22, 39, 41, 55]. However, they usually lack of geometry details and only capture the underlying human body, thus, not applicable for editing garments. Moreover, when applied to videos, visual artifacts of editing become more salient since they are not aware of underlying deformation of the garment's 3D geometry [25, 57].

We design our method to enjoy the advantages of both two approaches: preserving realistic details in UV mapping while circumventing 3D reconstruction. Our key insight is that the fundamental geometric property of isometry can be imposed into UV map estimation via the 3D surface normals predicted from an image. We formulate a geometric relationship between the UV map and surface normals in the form of a set of partial differential equations.

Our method takes as input an image or video, its surface normal prediction, and dense optical flow (for video), and outputs the geometry aware UV map estimate. The UV map is modeled by a multi-layer perceptron that can predict UV coordinates given a pixel location in an image. We note that the UV map is defined up to the choice of a reference coordinate frame. To disambiguate this, we condition the neural network with a pre-defined proxy UV map (e.g., DensePose [18]). We use the isometry constraints as a loss to optimize the UV map. Further, for a video, we leverage the per-frame image feature to correlate the UV coordinates of the pixels across time using optical flow.

Our contributions can be concluded in three aspects: (1) a novel formulation that captures the geometric relationship between the 3D surface normals and the UV map by the isometry contraint, which eliminates the requirement of 3D reconstruction and ground truth UV map; (2) a neural network design that learns to predict temporally coherent UV map for the frames by correlating per-frame image features; (3) stronger performance compared to existing re-texturing methods and compelling results on a wide range of real-world imagery.

2. Related Work

Our work lies at the intersection of human UV map prediction from images and neural UV map optimization.

2.1. Human Dense UV Map Estimation

A seminal work of DensePose [18] learns to predict a UV map of humans presented in an image, which opens a new opportunity to edit the appearance of a person without 3D reconstruction [3, 52]. A series of subsequent works [18, 28, 29, 39–42, 56, 58, 60] bring out a number of applications for human tracking. However, due to their representation specific to the body surface, they exhibit fundamental limitations in expressing highly deformable loose clothing such as skirts and dresses.

To address this challenge, recent approaches leverage multitask learning [61] or incorporate geodesic distance to learn UV maps [47]. BodyMap [22] incorporates the Vision Transformers to learn per-pixel image features on a continuous body surface that handles loose clothes, different hairstyles, and occlusion. TemporalUV [55] focuses on handling garments by extrapolating the initial DensePose estimates [18] and leveraging image features obtained from an input video to obtain a UV aligned with the garment boundary. Despite their promise, the visual artifacts persist due to a lack of understanding the underlying 3D geometry. Unlike previous approaches, we design our framework such that the resulting UV map satisfies the fundamental geometric property of isometry, which results in physically plausible re-texturing.

2.2. Neural UV Optimization from Videos.

Another line of work [24, 25, 34, 44, 57] resorts to a layered UV map, capturing the geometry to some degree by incorporating video decomposition [7, 8, 14, 30, 51] to optimize the UV coordinates of the foreground and background based on the observed motion. Kasten et al. [25] unwrap a video into a set of layered 2D atlases where for each pixel in the video, its corresponding 2D coordinate in each of the atlases is predicted. Ye et al. [57] proposes a global sprite image that can group the distinct motion trajectories because the collective object structure has a consistent appearance throughout time. While preserving the temporal coherency and maintaining some coarse UV deformations related to arm or leg movements during a sequence, these methods fall short of capturing micro deformations like wrinkles in

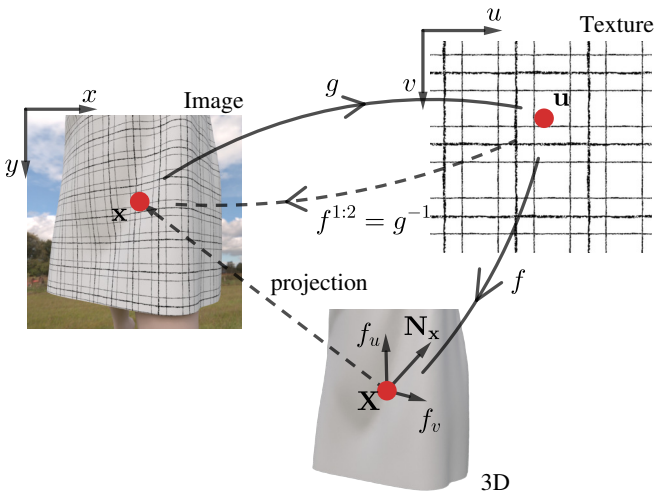

Figure 2. **Texture mapping geometry.** We study the mapping between the image space, texture space, and 3D space. A point $\mathbf{x}$ in image is mapped to the texture space with $\mathbf{u} = g(\mathbf{x})$. The mapping $f(\mathbf{u}) = \mathbf{X}$ lifts the texture plane into a 3D garment surface by an isometric warping. We use the orthographic projection model, resulting in the first two elements of f is the inverse of g, i.e., $f^{1:2} = g^{-1}$. The spatial derivatives of f form a tangent plane on the 3D surface at $\mathbf{X}$, resulting in $\mathbf{N}_\mathbf{x} = f_u \times f_v$ where $\mathbf{N}_\mathbf{x}$ is the 3D surface normal, and f_u and f_v are the spatial derivatives of f with respect to u and v.

the clothing. Furthermore, these techniques cannot represent UV mapping for an image and can only be applied to videos. Unlike these methods, our surface normal conditioned UV map is highly sensitive to small geometric details and 3D surface deformations, which can be optimized not only for a video but also for single images.

3. Method

Our goal is to obtain a continuous texture mapping, which allows editing the appearance of dynamic garments. We leverage the geometric property of isometry to constrain the UV map in the form of partial differential equations. We solve this partial differential equations by optimizing a neural network to generate a geometry aware UV map.

3.1. Texture Mapping without 3D Reconstruction

Consider a mapping $g(\mathbf{x}) = \mathbf{u}$ that maps a pixel location $\mathbf{x} = (x, y) \in \mathbb{R}^2$ in the image space that belongs to a garment of interest to a point $\mathbf{u} = (u, v) \in \mathbb{R}^2$ in the UV space of the garment as shown in Figure 2. The goal of our work is to find such a mapping g that takes into account the local surface geometry measured by the surface normal predicted at $\mathbf{x}$. We denote the predicted 3D surface normal of $\mathbf{x}$ in the camera space as $\mathbf{N}_\mathbf{x} \in \mathbb{S}^2$.

Let us define an *isometric* map from the UV texture map to the 3D surface, $f(\mathbf{u}) = \mathbf{X}$. This is the fundamental property of a non-stretchable cloth texture mapping [10].

$$\|f_u\| = \|f_v\| = 1, \quad f_u^\mathsf{T} f_v = 0, \tag{1}$$

where f_u and f_v are the partial derivatives of f with respect to u and v, respectively. Geometrically, f_u and f_v are the tangential vectors on the 3D surface where their cross product forms the surface normal:

$$\widetilde{\mathbf{N}}_\mathbf{x} = f_u \times f_v = f_u(g(\mathbf{x})) \times f_v(g(\mathbf{x})) \tag{2}$$

where $\widetilde{\mathbf{N}}_\mathbf{x} \in \mathbb{S}^2$ is the surface normal at $\mathbf{X}$ corresponding to $\mathbf{x}$.

We can find the UV mapping g by matching the surface normal $\widetilde{\mathbf{N}}_\mathbf{x}$ derived by Equation (2) and the surface normal predicted from the image $\mathbf{N}_\mathbf{x}$:

$$\underset{\boldsymbol{\theta}_g, \boldsymbol{\theta}_f}{\text{minimize}} \sum_\mathbf{x} \left\| f_u(g(\mathbf{x})) \times f_v(g(\mathbf{x})) - \mathbf{N}_\mathbf{x} \right\|^2, \quad \text{s.t.} \quad \|f_u\| = \|f_v\| = 1, f_u^\mathsf{T} f_v = 0, \tag{3}$$

where $\boldsymbol{\theta}_g$ and $\boldsymbol{\theta}_f$ are the parameters of the function g and f, respectively. A key challenge of solving Equation (3) lies in the dependency of f that requires full 3D reconstruction of the surface. Instead, we formulate a new dual problem that can solve Equation (3) effectively without finding f.

We use two properties to eliminate f from Equation (3). First, we assume orthographic projection, i.e., $(x, y) = (X, Y)$ where $\mathbf{X} = \begin{bmatrix} X & Y & Z \end{bmatrix}^\mathsf{T}$. This allows us to express the 3D derivatives using the pixel coordinates:

$$g = \left(f^{1:2}\right)^{-1}, \tag{4}$$

where $f^{1:2}$ is the first two elements (X, Y) of f. To keep $f^{1:2}$ bijective, we assume there is no self occlusion in the camera projection of f. Note that g^{-1} is the inverse of g that maps the UV texture map to the pixel coordinate. Second, we derive the derivatives of g by using the inverse function theorem [9]:

$$f^{1:2} \circ g(\mathbf{x}) = \mathbf{x} \;\; \rightarrow \;\; \mathbf{J}_g = \left(\mathbf{J}_{(f^{1:2})}\right)^{-1}, \tag{5}$$

where $\mathbf{J}_g = \begin{bmatrix} g_x & g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{bmatrix}$ is the Jacobian matrix of the function g.

With Equation (1) and (5), Equation (2) can be re-written as the following constraints by eliminating f (3D reconstruction):

$$\|g_x\| = \sqrt{1 + \frac{\widetilde{n}_x^2}{\widetilde{n}_z^2}}, \;\; \|g_y\| = \sqrt{1 + \frac{\widetilde{n}_y^2}{\widetilde{n}_z^2}}, \;\; g_x^\mathsf{T} g_y = \frac{\widetilde{n}_x \widetilde{n}_y}{\widetilde{n}_z^2}, \tag{6}$$

where $\widetilde{\mathbf{N}}_\mathbf{x} = \begin{bmatrix} \widetilde{n}_x & \widetilde{n}_y & \widetilde{n}_z \end{bmatrix}^\mathsf{T}$. For the derivation of Equation (6), see Supplementary Material.

Equation (6) is a set of partial differential equations of g that needs to match with the predicted surface normal $\mathbf{N}_\mathbf{x} =$

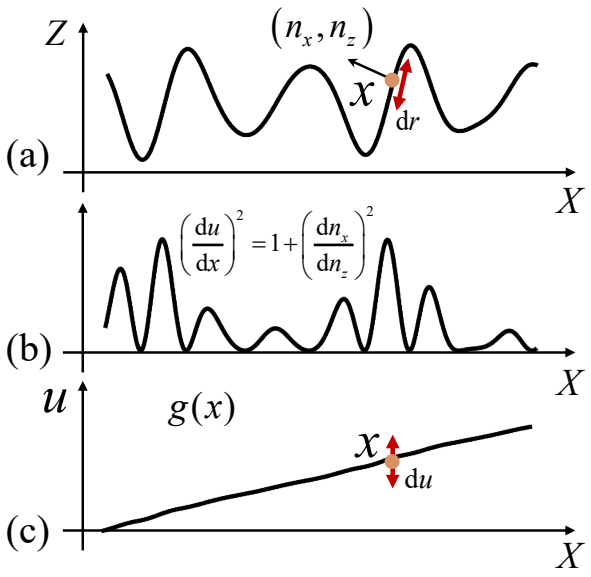

Figure 3. **2D simplification of UV map estimation.** We illustrate the isometric relationship between (a) the surface and (c) the UV map. The length of the curve in XZ axis needs to be preserved when mapping to $g(x)$, i.e., $|\mathrm{d}u| = |\mathrm{d}r| = \sqrt{\mathrm{d}x^2 + \mathrm{d}z^2}$ (arc length preservation). This relationship can be re-written as (b) a partial differential equation using Equation (6) in terms of its surface normal and the inverse of the spatial derivative of g, i.e., $\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)^2 = 1 + \left(\frac{\mathrm{d}n_x}{\mathrm{d}n_z}\right)^2$.

$\begin{bmatrix} n_x & n_y & n_z \end{bmatrix}^\mathsf{T}$, leading to a loss function:

$$\mathcal{L}_{\mathrm{geo}}(\boldsymbol{\theta}_g) = \sum_{\mathbf{x}} \left(\left(\frac{\partial u}{\partial x}\Big|_{\mathbf{x}} \right)^2 + \left(\frac{\partial v}{\partial x}\Big|_{\mathbf{x}} \right)^2 - 1 - \frac{n_x^2}{n_z^2} \right)^2 + \left(\left(\frac{\partial u}{\partial y}\Big|_{\mathbf{x}} \right)^2 + \left(\frac{\partial v}{\partial y}\Big|_{\mathbf{x}} \right)^2 - 1 - \frac{n_y^2}{n_z^2} \right)^2 + \left(\frac{\partial u}{\partial x}\Big|_{\mathbf{x}} \frac{\partial u}{\partial y}\Big|_{\mathbf{x}} + \frac{\partial v}{\partial x}\Big|_{\mathbf{x}} \frac{\partial v}{\partial y}\Big|_{\mathbf{x}} - \frac{n_x n_y}{n_z^2} \right)^2, \quad (7)$$

where $\frac{\partial u}{\partial x}\big|_{\mathbf{x}}$ is the partial derivative of u with respect to x evaluated at $\mathbf{x}$.

Figure 3 illustrates a 2D simplification of UV map estimation. A curve in XZ plane forms a isometric relation with the UV map, $|\mathrm{d}u| = |\mathrm{d}r| = \sqrt{\mathrm{d}x^2 + \mathrm{d}z^2}$. This relationship can be re-written as a partial differential equation in terms the surface normal (n_x, n_z) and the inverse of the spatial derivative of $g(x)$ using Equation (6), i.e., $\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)^2 = 1 + \left(\frac{\mathrm{d}n_x}{\mathrm{d}n_z}\right)^2$. We solve these partial differential equations to estimate g.

3.2. Self-supervised Learning of Texture Mapping

The texture map g is defined up to a bijective function, i.e., there exists an infinite number of g that are equivalent: $g^{-1} \circ g = (\mathcal{T} \circ g)^{-1} \circ (\mathcal{T} \circ g)$, where $\mathcal{T}$ is a bijective map (e.g., Euclidean transform). We resolve this ambiguity by finding g such that $g \approx g'$ where g' is a pre-defined proxy map of humans:

$$\mathcal{L}_{\mathrm{prox}}(\boldsymbol{\theta}_g) = \sum_{\mathbf{x}} \|g'(\mathbf{x}) - g(\mathbf{x})\|^2. \quad (8)$$

In practice, we use an extended DensePose [18] as the pre-defined proxy map. Since DensePose makes predictions only for the human body, we apply an extrapolation method [50] to inpaint the garment regions that are not covered by DensePose.

Further, we ensure physical plausibility of the visible 3D surfaces, i.e., the texture map should result in the surface normals pointing to $+Z$ direction, by adding the following loss:

$$\mathcal{L}_{\mathrm{z}}(\boldsymbol{\theta}_g) = \sum_{\mathbf{x}} \max(0, \det(\mathbf{J}_g|_{\mathbf{x}})), \quad (9)$$

where $\det(\mathbf{J}_g)$ is the determinant of the Jacobian $\mathbf{J}_g$ that is equivalent to $\widetilde{n}_z$. $\mathbf{J}_g|_{\mathbf{x}}$ is the Jacobian matrix of g evaluated at $\mathbf{x}$. See Supplementary Material for derivation.

For a video, we extend the texture map to include the image feature for each pixel, i.e., $g(\mathbf{x}, \mathbf{f}_{\mathbf{x}})$ where $\mathbf{f}_{\mathbf{x}}$ is the image feature at $\mathbf{x}$. This allows us to generalize the texture map over time. With the extension, we ensure the temporal consistency of the texture map by leveraging optical flow across frames:

$$\mathcal{L}_{\mathrm{tmp}}(\boldsymbol{\theta}_g) = \sum_{i,j} \sum_{\mathbf{x}_i} \left\| g(\mathbf{x}_i, \mathbf{f}_{\mathbf{x}_i}) - g\left(\mathbf{x}_j, \mathbf{f}_{\mathbf{x}_j}\right) \right\|^2, \quad (10)$$

where $\mathbf{x}_i$ is a point in the i^{th} frame. This point is mapped to $\mathbf{x}_j$ in the j^{th} frame, i.e., $\mathbf{x}_j = W_{i \to j}(\mathbf{x}_i)$ where $W_{i \to j}$ is the optical flow from the i^{th} to j^{th} frames.

Overall, we optimize the following loss to learn the texture map:

$$\mathcal{L}(\boldsymbol{\theta}_g) = \mathcal{L}_{\mathrm{geo}} + \lambda_{\mathrm{prox}} \mathcal{L}_{\mathrm{prox}} + \lambda_{\mathrm{z}} \mathcal{L}_{\mathrm{z}} + \lambda_{\mathrm{tmp}} \mathcal{L}_{\mathrm{tmp}}, \quad (11)$$

where λ_{prox}, λ_{z}, and λ_{tmp} are the weights that determine the relative importance of losses. Note that when a single image is used, $\lambda_{\mathrm{tmp}} = 0$.

3.3. Implementation Details

We model g using a 12-layer multi-layer perception, with ReLU [2] as an activation function after each layer that takes as input a pixel coordinate with positional encoding, $\gamma(\mathbf{x})$ where $\gamma : \mathbb{R}^2 \to \mathbb{R}^{128}$ is Fourier based positional encoding [48]. For videos, we use ResNet [19] to extract per-frame 256 dimensional image feature $\mathbf{f}_{\mathbf{x}}$. Our network design is illustrated in Figure 4 (for image-based UV map prediction) and 5 (for video-based UV map prediction). To make our prediction scale-invariant, we crop the garments region with 256×256 resolution. We use an off-the-shelf garment segmentation software, Graphonomy [15]

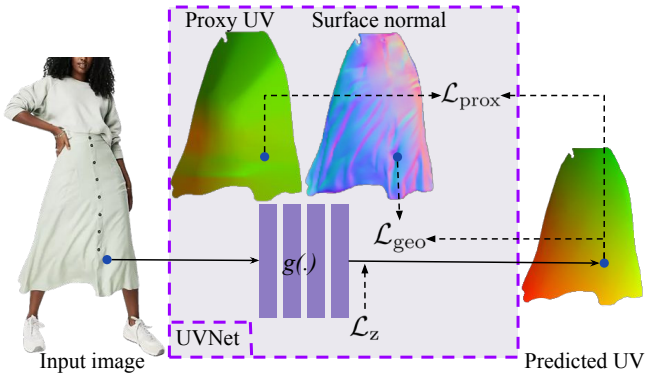

Figure 4. **Single image framework.** For each pixel location on a garment $\mathbf{x}$, we predict the UV coordinate $\mathbf{u}$ using a multi-layer perceptron. We enforce isometry to the UV map by matching with the predicted 3D surface normals. $\mathcal{L}_{\text{z}}$ ensures that the predicted surface normals point toward the camera. To disambiguate the frame of reference of the UV map, we use pre-defined proxy map (e.g., DensePose [18] extrapolation [50]) by applying $\mathcal{L}_{\text{prox}}$.

to separate the garment area. We use Adam optimizer [27] with batch size of 2048 and learning rate of 10^{-4}. We set $\lambda_{\text{prox}} = 0.2$, $\lambda_{\text{z}} = 0.01$, and $\lambda_{\text{tmp}} = 0.3$. We used an NVIDIA V100 GPU and Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20GHz, and implemented our approach with Pytorch [43]. Our method takes 20 minutes for a video of 82 frames while Kasten et al. [25], Ye et al. [57], and TemporalUV [55] take 10 hours, 30 minutes, and 23 hours, respectively.

4. Evaluation

We evaluate our method both quantitatively and qualitatively on real images as well as synthetic data with ground truth UV map. We also compare with the state-of-the-art methods on human UV map estimation methods and video UV map optimization approaches.

Evaluation Datasets We evaluate our method using the following datasets: (1) five synthetic video sequences of simulated dress and T-shirt garments from Santesteban et al. [45] with random texture patterns over 700 frames; (2) ten real videos from Fashion Video dataset [59]; (3) TikTok dataset [23] and various YouTube videos as well as in-the-wild internet images.

Evaluation Metric We use five metrics to evaluate our method. (1) UV error: for synthetic data, we report the absolute UV error in the texture space [45]. We use a Procrustes analysis [16] to align the resulting texture map to account for the diambiguity of the reference frame. We report the mean squared error in metric scale by assuming the height of the person in the input is 165cm, resulting in 0.41cm/UV for dress and 0.27cm/UV for T-shirt (Table 1). (2) Average Precision percentage: we report the Average Precision (AP) percentage metric computed on all the pixels considering a per-pixel prediction as correct if the UV error is lower than a threshold. We visualize the AP metric for a range of thresholds from 1 to 15 cm and obtain the graph shown in Figure 7. (3) Photometric error: we warp the first frame of an input video to the rest of the frames using the UV map estimates. We report the error between the ground truth RGB images and the warped RGB images as reported in Table 1. (4) Geometric error: we report the $\mathcal{L}_{\text{geo}}$ to show how the predicted UV map follows the geometric information captured in the surface normal estimates as reported in Table 1. (5) Temporal error: we evaluate the capability of the different approaches in preserving the temporal coherency by reporting the $\mathcal{L}_{\text{tmp}}$ error (Table 1). When the ground truth UV is not available, we use the geometric and temporal errors to evaluate our method.

Baseline Methods We compare our method with previous works that fall into two categories: (1) human UV map prediction; (2) UV optimization.

1) *Human UV map prediction:* we compare our method with state-of-the-art that focus on predicting UV maps for the naked human body [18] and dressed humans [47, 55]. We also report the performance of the UV map obtained by extrapolating DensePose predictions as discussed in Section 3.3. Our method achieves the best performance as shown in Table 1 and Figure 7. We notice that DensePose [18] performs competently in precision percentage when the threshold error is less than 7 cm. This observation is based on the fact that, for each pixel, DensePose predicts a part label (among 24 parts) and a UV map with respect to that body part. When aligning these predictions with the ground truth, we warp each occupied body part individually, resulting in a more accurate alignment compared to the other methods (including ours) that are represented by only one patch. However, the performance of DensePose [18] is not improved above 7cm because of limited ability to predict beyond body surface.

2) *UV optimization:* we compare our method with state-of-the-art in predicting the UV map of a dynamic object observed in a video [25, 57]. However, such methods are not tailored for garments that undergo highly non-linear transformations as the body moves. Hence, as reported in Table 1 and Figure 7, our method surpasses these baselines in the dense UV error, the average precision percentage, the geometric error, and the temporal error.

Ablation Study We conduct an ablation study to analyze the impact of the distance (first two terms in Equation (7)) and angle (third term in Equation (7)) constraints (second and third rows of Table 2 and Figure 7). Our final method has the best performance in UV error and photometric error. We also compare the performance of our method without the temporal consistency ($\mathcal{L}_{\text{tmp}}$) (fourth row of Table 2). As expected, this term performs quite similarly to ours in UV error but very poorly in photometric error since the consistency between the frames is not enforced. The role of

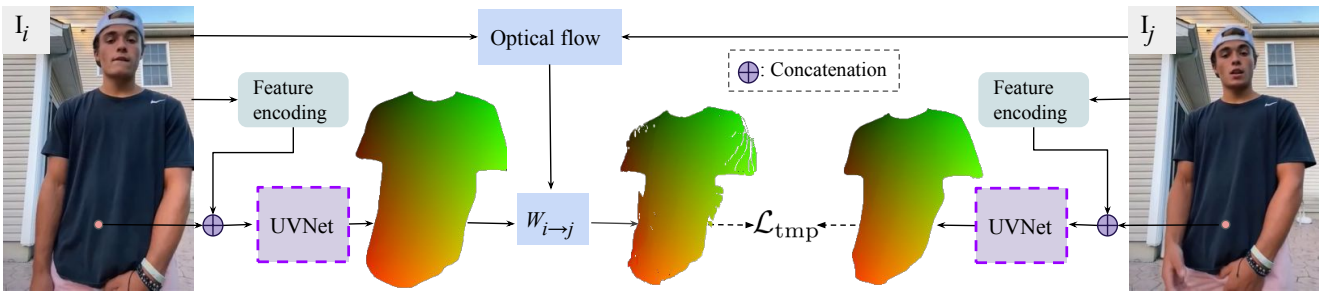

Figure 5. **Video framework.** We apply temporal coherence in the predicted UV maps using optical flow [35, 36, 49]. With the optical flow $W_{i\to j}$, we match the UV prediction of i^{th} frame with j^{th} via $\mathcal{L}_{\text{tmp}}$.

	GT dress sequences [45]		GT T-shirt sequences [45]		Real Fashion sequences [59]	
Method	UV. error (cm)	photo. error	UV. error (cm)	photo. error	geo. error ($\mathcal{L}_{\text{geo}}$)	tmp. error
DensePose [18]	17.27±3.76	52.43±10.20	7.48±0.52	34.19±7.01	1.26±0.47	8.12±2.70
Extrapolated DensePose [18, 50]	8.61±0.76	18.29±3.24	5.34±0.63	18.86±4.55	0.52±0.06	4.62±0.73
HumanGPS [47]	11.97±2.06	97.41±30.87	7.53±0.43	108.10±34.89	1.27±0.69	51.40±41.66
Kasten et al. [25]	7.06±0.60	13.07±2.79	6.64±0.63	14.54±4.26	0.56±0.10	2.30±0.70
Ye et al. [57]	5.56±0.29	33.57±-10.27	5.75±0.20	19.22±5.14	0.71±0.03	1.65±0.15
Ours	**3.16±0.28**	**7.54±2.04**	**3.58±0.27**	**11.28±2.01**	**0.07±0.03**	**1.50±0.23**

Table 1. Quantitative Results. UV. error (cm), photo. error (RGB difference), geo. error ($\mathcal{L}_{\text{geo}}$), and tmp. error ($\mathcal{L}_{\text{tmp}}$) (image space pixel distance) respectively (mean±std).

Method	UV. error (cm)	photo. error (RGB)
Ours	**3.16±0.28**	**7.54±2.04**
Distance constraint	3.44±0.25	8.29±1.99
Angle constraint	5.96±0.33	7.68±1.37
W/o $\mathcal{L}_{\text{tmp}}$	3.22±0.24	14.67±4.82
W/o $\mathcal{L}_{\text{prox}}$	3.24±0.35	8.46±2.19

Table 2. Ablation study on dress sequences [45]. UV. error (cm), photo. error (RGB difference) (mean±std).

Figure 6. The impact of $\mathcal{L}_{\text{prox}}$ in UV prediction and retexturing.

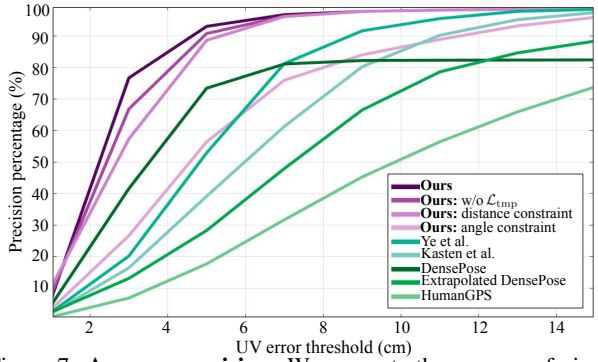

Figure 7. **Average precision.** We compute the average of pixels with a UV error higher than a threshold (1-15cm) on our method, the baseline methods, and our ablation experiments on ground truth dress sequences [45].

$\mathcal{L}_{\text{prox}}$ is the disambiguation of UV maps, i.e., there exist an infinite number of equivalent UV maps that minimize our $\mathcal{L}_{\text{geo}}$ (PDE). While the result without $\mathcal{L}_{\text{prox}}$ are, therefore, quantitatively competitive as summarized in Table 2, such ambiguity can be resolved by finding UV that is closest to the proxy UV as shown in Figure 6, i.e., the retexture without $\mathcal{L}_{\text{prox}}$ can result in arbitrary orientation across subjects.

Qualitative Results To show our results qualitatively, we visualize both re-texturing examples and grid UV illustration to depict the performance of each method in preserving high-frequency details. For retexturing, we first obtain an albedo and shading layer from the input image using an intrinsic image decomposition method [53]. After applying a new texture pattern to the albedo layer, we composite it back with the original shading layer. We apply Gamma-correction [37] on the input image, after generating the albedo layer and shading layer, we inverse the Gamma-correction back when synthesizing the re-textured image. We compare our method qualitatively with the baselines as shown in Figure 8 that illustrates the results on Fashion video sequence [59]. Figure 11 shows the performance of our method on videos and images. Our method not only captures the fine-grained surface details but also is temporally coherent across time.

User Study We conducted a user study: asking participants ($n = 21$) to rate realism (1: unrealistic to 10: most realistic) for our method compared to the baselines as summarized in Table 3. Our method receives the highest score from the users.

Method	User score
DensePose	4.09±1.94
Proxy DensePose	3.23±1.86
HumanGPS	2.28±2.12
Kasten et al.	5.52±1.88
Ye et al.	7.66±1.27
Ours	**9.57±0.81**

Table 3. User study.

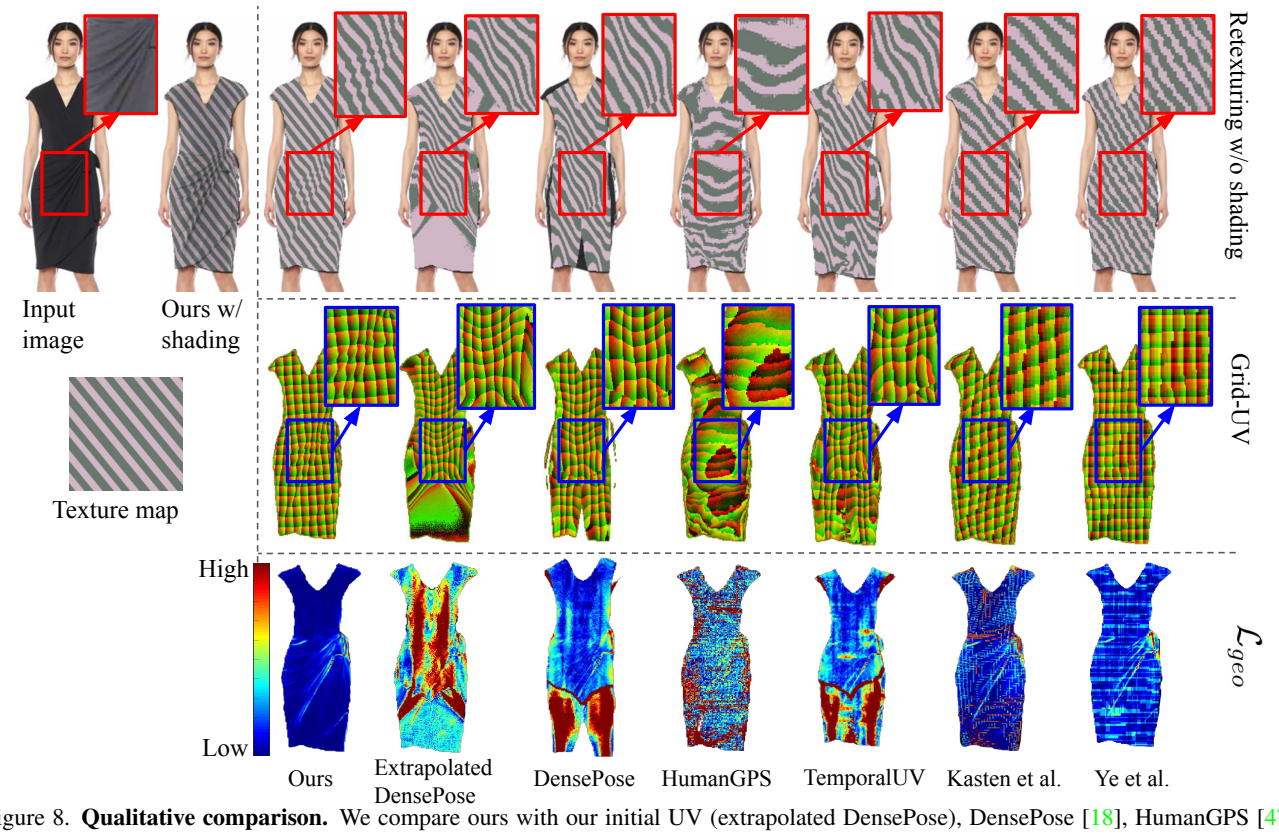

Figure 8. **Qualitative comparison.** We compare ours with our initial UV (extrapolated DensePose), DensePose [18], HumanGPS [47], Kastenn et al. [25], and Ye et al. [57] in image re-texturing and UV map grid visualization.

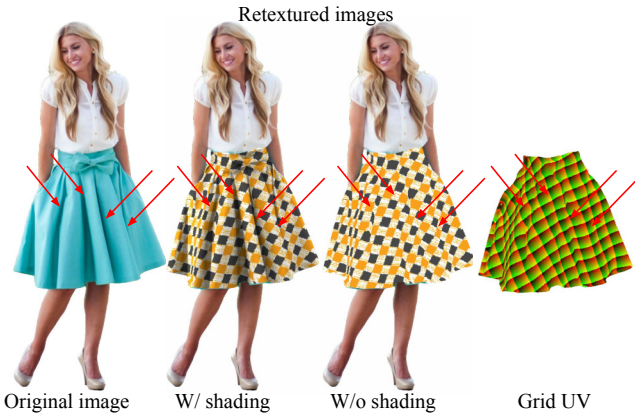

Figure 9. **Limitation.** Due to lack of 3D reconstructed surface, our method cannot handle folds that introduce texture discontinuity. Our texture map is continuous around the folds, which is physically incorrect.

5. Discussion

This paper presents a novel approach to predict a high quality UV map by preserving geometric details from images and videos. We leverage the geometric property of isometry encoded in 3D surface normals to optimize the UV map in the form of partial differential equations. We generalize our method to videos by integrating optical flow, resulting in a temporally coherent video editing. Our method produces strong qualitative and quantitative predictions on real-world imagery compared to state-of-the-art UV map estimation.

Limitation As discussed in Section 3.1, our method makes an assumption about projection, i.e., there is one-to-one correspondence between 3D surface geometry and image. However, this assumption does not hold when there is a fold where a region of 3D surface is not visible to the image. This makes a contrast with 3D reconstruction based method where the invisible part of 3D surface can be still mapped to the image via depth reasoning. Figure 9 illustrates this limitation where there are folds in the skirt, resulting in negative surface normal n_z. Due to the folds, the texture must be discontinuous while our method produces continuous texture rendering due to the lack of 3D reconstructed geometry, which is physically incorrect. Our pipeline is composed of two components: (1) UV prediction (our contribution) and (2) texture map with shading (not our contribution). For the garments with highly contrasted texture, the shading operation is often biased to color contrast, resulting in erroneous appearance. Figure 10 illustrates that despite reasonable UV prediction from our method, the resulting appearance is unrealistic near the textured region. Improving the shading operation is beyond the scope of this work, and we leave it as future work.

Figure 10. **Limitation.** Limitations on textured garment.

Acknowledgement This work was supported by a NSF NRI 2022894 and NSF CAREER 1846031.

(a) Qualitative results of our method on an image.

(b) Qualitative results of our method on videos.

Figure 11. **Results on images and videos.** Our method can produce compelling texture editing given images and videos that preserve geometric details.

References

[1] https://renderpeople.com/3d-people. 2

[2] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv*, 2018. 4

[3] Badour AlBahar, Jingwan Lu, Jimei Yang, Zhixin Shu, Eli Shechtman, and Jia-Bin Huang. Pose with Style: Detail-preserving pose-guided image synthesis with conditional stylegan. *TOG*, 2021. 2

[4] Yazeed Alharbi and Peter Wonka. Disentangled image generation through structured noise injection. In *CVPR*, 2020. 2

[5] David Bau, Hendrik Strobelt, William Peebles, Jonas Wulff, Bolei Zhou, Jun-Yan Zhu, and Antonio Torralba. Semantic photo manipulation with a generative image prior. *TOG*, 2019. 2

[6] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *ICCV*, 2019. 2

[7] Michael J. Black and P. Anandan. Robust dynamic motion estimation over time. In *CVPR*, 1991. 2

[8] Gabriel J Brostow and Irfan A Essa. Motion based decompositing of video. In *ICCV*, 1999. 2

[9] Felix E. Browder. On the unification of the calculus of variations and the theory of monotone nonlinear operators in banach spaces. *PNAS*, 1966. 3

[10] Edwin Earl Catmull. *A subdivision algorithm for computer display of curved surfaces*. The University of Utah, 1974. 3

[11] Shu-Yu Chen, Wanchao Su, Lin Gao, Shihong Xia, and Hongbo Fu. Deepfacedrawing: Deep generation of face images from sketches. *TOG*, 2020. 2

[12] Anton Cherepkov, Andrey Voynov, and Artem Babenko. Navigating the gan parameter space for semantic image editing. In *CVPR*, 2021. 2

[13] Edo Collins, Raja Bala, Bob Price, and Sabine Süsstrunk. Editing in style: Uncovering the local semantics of GANs. In *CVPR*, 2020. 2

[14] T. Darrell and A. Pentland. Robust estimation of a multi-layered motion representation. In *Proceedings of the IEEE Workshop on Visual Motion*, 1991. 2

[15] Ke Gong, Yiming Gao, Xiaodan Liang, Xiaohui Shen, Meng Wang, and Liang Lin. Graphonomy: Universal human parsing via graph transfer learning. In *CVPR*, 2019. 4

[16] J.C. Gower and G.B. Dijksterhuis. Procrustes problems. new york: Oxford university press. *Psychometrika*, 70, 2005. 5

[17] Artur Grigorev, Artem Sevastopolsky, Alexander Vakhitov, and Victor Lempitsky. Coordinate-based texture inpainting for pose-guided human image generation. In *CVPR*, 2019. 2

[18] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018. 2, 4, 5, 6, 7

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[20] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want. *TIP*, 2019. 2

[21] Xianxu Hou, Xiaokang Zhang, Linlin Shen, Zhihui Lai, and Jun Wan. Guidedstyle: Attribute knowledge guided style manipulation for semantic face editing. *arXiv*, 2020. 2

[22] Anastasia Ianina, Nikolaos Sarafianos, Yuanlu Xu, Ignacio Rocco, and Tony Tung. Bodymap: Learning full-body dense correspondence map. In *CVPR*, 2022. 2

[23] Yasamin Jafarian and Hyun Soo Park. Learning high fidelity depths of dressed humans by watching social media dance videos. In *CVPR*, 2021. 5

[24] Varun Jampani, Raghudeep Gadde, and Peter V. Gehler. Video propagation networks. In *CVPR*, 2017. 2

[25] Yoni Kasten, Dolev Ofri, Oliver Wang, and Tali Dekel. Layered neural atlases for consistent video editing. *TOG*, 2021. 2, 5, 6, 7

[26] Hyunsu Kim, Yunjey Choi, Junho Kim, Sungjoo Yoo, and Youngjung Uh. Exploiting spatial dimensions of latent in gan for real-time image editing. In *CVPR*, 2021. 2

[27] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[28] Nilesh Kulkarni, Abhinav Gupta, David F. Fouhey, and Shubham Tulsiani. Articulation-aware canonical surface mapping. In *CVPR*, 2020. 2

[29] Nilesh Kulkarni, Abhinav Gupta, David F Fouhey, and Shubham Tulsiani. Articulation-aware canonical surface mapping. In *CVPR*, 2020. 2

[30] M. Pawan Kumar, Philip H. S. Torr, and Andrew Zisserman. Learning layered motion segmentations of video. *IJCV*, 2005. 2

[31] Verica Lazova, Eldar Insafutdinov, and Gerard Pons-Moll. 360-degree textures of people in clothing from a single image. *3DV*, 2019. 2

[32] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *CVPR*, 2020. 2

[33] Huan Ling, Karsten Kreis, Daiqing Li, Seung Wook Kim, Antonio Torralba, and Sanja Fidler. Editgan: High-precision semantic image editing. In *NeurIPS*, 2021. 2

[34] Erika Lu, Forrester Cole, Tali Dekel, Weidi Xie, Andrew Zisserman, David Salesin, William T. Freeman, and Michael Rubinstein. Layered neural rendering for retiming people in video. *TOG*, 2020. 2

[35] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI*, 1981. 6

[36] Bruce D. Lucas and Takeo Kanade. Optical navigation by the method of differences. In *IJCAI*, 1985. 6

[37] Tom McReynolds and David Blythe. *Advanced graphics programming using OpenGL*. Elsevier, 2005. 6

[38] Aymen Mir, Thiemo Alldieck, and Gerard Pons-Moll. Learning to transfer texture from clothing images to 3d humans. In *CVPR*, 2020. 2

[39] Natalia Neverova, David Novotny, Vasil Khalidov, Marc Szafraniec, Patrick Labatut, and Andrea Vedaldi. Continuous surface embeddings. In *NeurIPS*, 2020. 2

[40] Natalia Neverova, David Novotny, and Andrea Vedaldi. Correlated uncertainty for learning dense correspondences from noisy labels. In *NeurIPS*, 2019. 2

[41] Natalia Neverova, Artsiom Sanakoyeu, David Novotny, Patrick Labatut, and Andrea Vedaldi. Discovering relationships between object categories via universal canonical maps. In *CVPR*, 2021. 2

[42] Natalia Neverova, James Thewlis, Riza Alp Güler, Iasonas Kokkinos, and Andrea Vedaldi. Slim densepose: Thrifty learning from sparse annotations and motion cues. In *CVPR*, 2019. 2

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*. 2019. 5

[44] Alex Rav-Acha, Pushmeet Kohli, Carsten Rother, and Andrew Fitzgibbon. Unwrap mosaics: A new representation for video editing. *SIGGRAPH*, 2008. 2

[45] Igor Santesteban, Nils Thuerey, Miguel A Otaduy, and Dan Casas. Self-Supervised Collision Handling via Generative 3D Garment Models for Virtual Try-On. In *CVPR*, 2021. 5, 6

[46] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *CVPR*, 2020. 2

[47] Feitong Tan, Danhang Tang, Dou Mingsong, Guo Kaiwen, Rohit Pandey, Cem Keskin, Ruofei Du, Deqing Sun, Sofien Bouaziz, Sean Fanello, Ping Tan, and Yinda Zhang. Humangps: Geodesic preserving feature for dense human correspondences. In *CVPR*, 2021. 2, 5, 6, 7

[48] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *NeurIPS*, 2020. 4

[49] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020. 6

[50] Alexandru Telea. An image inpainting technique based on the fast marching method. *Journal of Graphics Tools*, 2004. 4, 5, 6

[51] John Y. A. Wang and Edward H. Adelson. Representing moving images with layers. *TIP*, 1994. 2

[52] Tuanfeng Wang, Duygu Ceylan, Krishna Kumar Singh, and Niloy J. Mitra. Dance in the wild: Monocular human animation with neural dynamic appearance synthesis. In *3DV*, 2021. 2

[53] Yair Weiss. Deriving intrinsic images from image sequences. In *ICCV*, 2001. 6

[54] Rongliang Wu, Gongjie Zhang, Shijian Lu, and Tao Chen. Cascade ef-gan: Progressive facial expression editing with local focuses. In *CVPR*, 2020. 2

[55] You Xie, Huiqi Mao, Angela Yao, and Nils Thuerey. Temporaluv: Capturing loose clothing with temporally coherent uv coordinates. In *CVPR*, 2022. 2, 5

[56] Haonan Yan, Jiaqi Chen, Xujie Zhang, Shengkai Zhang, Nianhong Jiao, Xiaodan Liang, and Tianxiang Zheng. Ultrapose: Synthesizing dense pose with 1 billion points by human-body decoupling 3d model. In *ICCV*, 2021. 2

[57] Vickie Ye, Zhengqi Li, Richard Tucker, Angjoo Kanazawa, and Noah Snavely. Deformable sprites for unsupervised video decomposition. In *CVPR*, 2022. 2, 5, 6, 7

[58] Zhixuan Yu, Haozheng Yu, Long Sha, Sujoy Ganguly, and Hyun Soo Park. Semi-supervised dense keypointsusing unlabeled multiview images. *arXiv*, 2021. 2

[59] Polina Zablotskaia, Aliaksandr Siarohin, Bo Zhao, and Leonid Sigal. Dwnet: Dense warp-based network for pose-guided human video generation. In *BMVC*, 2019. 5, 6

[60] Wang Zeng, Wanli Ouyang, Ping Luo, Wentao Liu, and Xiaogang Wang. 3d human mesh regression with dense correspondence. In *CVPR*, 2020. 2

[61] Tyler Zhu, Per Karlsson, and Chris Bregler. Simpose: Effectively learning densepose and surface normal of people from simulated data. In *ECCV*, 2020. 2