Texture Generation Using Dual-Domain Feature Flow with Multi-View Hallucinations

Seunggyu Chang[1], **Jungchan Cho**[2], **Songhwai Oh**[1]
[1]Department of ECE, ASRI, Seoul National University
[2]School of Computing, Gachon University
seunggyu.chang@rllab.snu.ac.kr, thinkai@gachon.ac.kr, songhwai@snu.ac.kr

Abstract

We propose a dual-domain generative model to estimate a texture map from a single image for colorizing a 3D human model. When estimating a texture map, a single image is insufficient as it reveals only one facet of a 3D object. To provide sufficient information for estimating a complete texture map, the proposed model simultaneously generates multi-view hallucinations in the image domain and an estimated texture map in the texture domain. During the generating process, each domain generator exchanges features to the other by a flow-based local attention mechanism. In this manner, the proposed model can estimate a texture map utilizing abundant multi-view image features from which multi-view hallucinations are generated. As a result, the estimated texture map contains consistent colors and patterns over the entire region. Experiments show the superiority of our model for estimating a directly render-able texture map, which is applicable to 3D animation rendering. Furthermore, our model also improves an overall generation quality in the image domain for pose and viewpoint transfer tasks.

Introduction

Along with the increase of online activities recently, reconstructing a 3D avatar from photos becomes an important problem. To reconstruct a 3D avatar, we need a 3D mesh template for shape representation and a corresponding texture map for colorization. Traditionally, a 3D avatar reconstruction requires multiple image pairs consisting of diverse poses and viewpoints taken at a dedicated studio. However, trend has move on to reconstructing a 3D model from fewer or a single image. Many works (Bogo et al. 2016; Lassner et al. 2017; Kanazawa et al. 2018; Pavlakos et al. 2018; Varol et al. 2018; Alldieck et al. 2019; Natsume et al. 2019; Weng, Curless, and Kemelmacher-Shlizerman 2019; Gabeur et al. 2019; Saito et al. 2019; Kolotouros et al. 2019; Choutas et al. 2020; Saito et al. 2020) have been studied to resolve 3D human shape reconstruction from a single image, however, little has been studied for texture map reconstruction from a single image for a 3D model (Jian et al. 2019; Lazova, Insafutdinov, and Pons-Moll 2019).

A texture map is an image lying on uv-coordinates containing whole surface colors of a 3D model (Catmull 1974; Catmull and Smith 1980). When rendering an image using the 3D model, the surface color values are drawn from the texture map according to the pre-defined uv parameterization. One hardness of the texture map estimation from a single image is that a single image is insufficient for generating a complete texture map for the entire surface. Conventionally, a texture map is obtained by aggregating multiple partial texture maps, obtained by unwrapping multi-view images into uv-coordinates, called stitching (Wang et al. 2001; Thormählen and Seidel 2008). However, when it comes to generating a texture map from a single image, a half of the entire surface is unseen, whose colors should be filled by imagination.

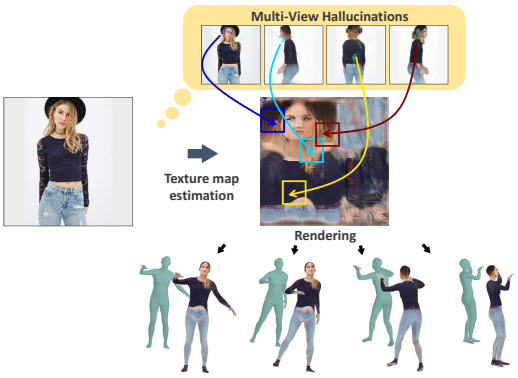

Figure 1: Estimating a texture map from a single image by hallucinating multi-view images. An arrow from each hallucination image represents a reference part for estimating a boxed region on the estimated texture map.

To resolve this issue, we propose a texture map estimation method from a single image for colorizing a 3D human model utilizing multi-view features. Recently, deep architectures have achieved remarkable success on pose transferred image generation, and we expect that deep architectures can provide sufficient alternatives for real multi-view images. To this end, instead of generating a texture map solely, we design our model to generate images in two different domains: pose transferred images in the image domain and a texture map in the texture domain. Our model consists of two main generators: an image generator and a texture generator. During the generation process, the image generator generates

pseudo multi-view images from a single image to provide an entire surface features for generating a texture map. Simultaneously, the texture generator generates the texture map utilizing features from the pseudo multi-view images, which we named as hallucinations. The overall texture generating scheme is depicted in Figure 1. The texture domain generator utilizes multi-view features drawn from the image generator and makes the image generator utilize texture features drawn from the texture domain generator using an attention mechanism. As a consequence, each domain's generator takes advantage of geometric clues and information about unseen surfaces. In light of the image domain generator, texture features provide geometrically consistent patterns over any pose, viewpoint, and scale, increasing generalization performance even for the out-of-distribution cases. From the texture domain generator's standpoint, image features provide natural colors and pattern information dedicated to generating real-looking images and clues for unseen surfaces.

To validate our method, we conduct experiments on various datasets and show the superiority of our method for generating a texture map and pose transferred images. We also demonstrate our resulting texture map can be applied to 3D human model for rendering a 3D animation clip.

We can summarize our contributions into three folds.

- We propose a novel multi-view hallucination generation scheme to provide pseudo multi-view images for estimating a texture map from a single image.
- We propose dual-domain generators in which each domain feature is interacting with the other by an attention mechanism, which improves generated image qualities for image pose transfer and texture map estimation tasks.
- We generate a directly render-able texture map in decent quality for the 3D human model from a single image.

Related Work

Texture map estimation. Densepose transfer (Neverova, Alp Guler, and Kokkinos 2018), which firstly adopts a texture mapping technique on pose transferred image generation, inpaints a partial texture map warped from an input image to generate a complete texture map. Grigorev et al. (2019) estimate a flow-field from a partial flow map by which an input image is warped to make a full texture map. However, their resulting texture maps have limited quality for being directly used for rendering, which requires additional processing layers to obtain a final pose transferred image. Jian et al. (2019) utilize person re-identification loss to generate a direct render-able texture map, albeit their results are somewhat blurry. Lazova, Insafutdinov, and Pons-Moll (2019) proposed texture and displacement-map-generating networks from a single image trained using full texture maps obtained from elaborately synthesized 3D models. The generated texture map by Lazova, Insafutdinov, and Pons-Moll (2019) has sufficient fidelity, however, training them requires full texture maps which are hard to obtain in practice. Utilizing texture mapping for pose transferred image generation has an advantage of keeping temporal consistency on the same surfaces when generating a video clip. Zhi et al. (2020) proposed a texture and displacement generating framework from multiple RGB-D frames of a video. Our approach differs from Zhi et al. (2020) in that we assume multi-view images for generating a texture map are not given as inputs, but an another objective to generate during the process.

Preliminary

Global flow local attention (GFLA) (Ren et al. 2020) is a patch-based attention module in which a local patch is extracted from where a flow points to. GFLA consists of two modules: *flow generator,*[1] F, and *local attention module*, A. A flow generator F generates a flow $\mathbf{f}$, according to which local patches are extracted[2], and a binary mask $\mathbf{m}$ for merging features. Let $\boldsymbol{\phi}^q$ and $\boldsymbol{\phi}^k$ denote a query and a key feature respectively, and $\mathbf{f}^{kq}$ denote a flow from key to query. The local attention module outputs $\boldsymbol{\phi}^{out} = A(\boldsymbol{\phi}^q, \boldsymbol{\phi}^k, \mathbf{f}^{kq})$ in two steps. Let $\mathcal{N}_n(\boldsymbol{\phi}, l)$ be an $n \times n$ sized local patch extracted from $\boldsymbol{\phi}$ centered at location l. In a local attention module, a local attention feature $\boldsymbol{\phi}^{attn}$ is computed as

$$\boldsymbol{\phi}^{attn}(l) = \text{Attn}\Big(\mathcal{N}_n(\boldsymbol{\phi}^q, l), \mathcal{N}_n\big(\boldsymbol{\phi}^k, l + \mathbf{f}^{kq}(l)\big)\Big), \quad (1)$$

where $\text{Attn}(\cdot, \cdot)$ is a general attention module (Vaswani et al. 2017). Then the final output $\boldsymbol{\phi}^{out}$ is computed as

$$\boldsymbol{\phi}^{out} = (\mathbf{1} - \mathbf{m}^{kq}) \otimes \boldsymbol{\phi}^q + \mathbf{m}^{kq} \otimes \boldsymbol{\phi}^{attn}. \quad (2)$$

where $\mathbf{m}^{kq}$ denotes a binary mask generated by F together with $\mathbf{f}^{kq}$, $\otimes$ denotes an element-wise multiplication, and $\mathbf{1}$ denotes a tensor whose elements are all ones.

Proposed Method

Notations. Let $\mathbf{x}$ denote an image, $\mathbf{s}$ denote a surface annotation representing texel[3] coordinates of pixels in uv, obtained by DensePose (Güler, Neverova, and Kokkinos 2018). Let $\mathbf{p}$ denote an image pose of the image $\mathbf{x}$ represented as a heat map of keypoints detected by OpenPose (Cao et al. 2019). Let $\mathbf{t}$ denote an estimated texture map and $\mathbf{c}$ denote a coordinate annotation representing pixel coordinates of texels. The coordinate annotation $\mathbf{c}$ and the surface annotation $\mathbf{s}$ are inversely related satisfying $l = \mathbf{c}(\mathbf{s}(l))$ for any pixel coordinate l on a human body. Let $\mathbf{b}$ denote a texture pose, a warped image pose $\mathbf{p}$ to the texture domain according to the coordinate annotation $\mathbf{c}$, namely $\mathbf{b} = warp(\mathbf{p}; \mathbf{c})$. Superscript s and t are used to denote source and target, identifying that a symbol is used for pre-/post-pose-transform, respectively, and h is also used in place of t to emphasize that targets are used for hallucination. Please refer to the supplementary material for more detailed notations.

Formulation. Our model consists of two generative network pipelines: a *hallucination network complex* (*hallunet-complex*, H-Nets$_{\mathcal{I}}$) and a *texture network complex* (*texnet-complex*, T-Nets$_{\mathcal{T}}$). Step I in Figure 2 depicts an overview

[1] It is called a *global flow field estimator* in the original paper.

[2] Originally, $\mathbf{f}$ represents relative positions, however, we use relative positional representation when key and query features lie on the same domain, and absolute positional representation elsewhere.

[3] pixel of a texture map

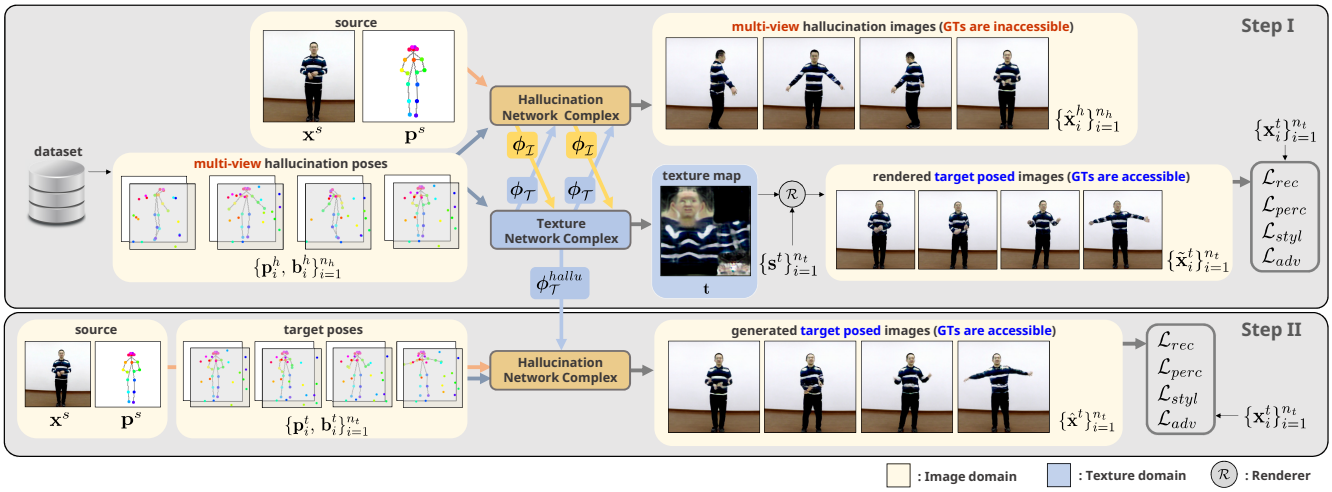

Figure 2: Overview of the proposed dual-domain generative method consisting of two generative pipelines: a *hallucination network complex (H-Nets$_\mathcal{I}$)* and a *texture network complex (T-Nets$_\mathcal{T}$)*. The H-Nets$_\mathcal{I}$ generates n_h pose-transferred hallucination images, $\{\hat{\mathbf{x}}_i^h\}_{i=1}^{n_h}$, on the image domain while the T-Nets$_\mathcal{I}$ generates an estimated texture map, $\mathbf{t}$, on the texture domain. The two pipelines processed simultaneously exchanging their features ($\phi_\mathcal{I}$: image feature, $\phi_\mathcal{T}$: texture feature) to the other domain.

of the generation process. The hallunet-complex, H-Nets$_\mathcal{I}$, simultaneously generates n_h pose transferred hallucination images, $\{\hat{\mathbf{x}}_i^h\}_{i=1}^{n_h}$, from a source image, $\mathbf{x}^s$, and n_h hallucination image poses, $\{\mathbf{p}_i^h\}_{i=1}^{n_h}$, while the texnet-complex, T-Nets$_\mathcal{T}$, generates an estimated texture map, $\mathbf{t}$. The hallunet-complex and the texnet-complex are processed simultaneously referring to intermediate features of the other domain by GFLA. To utilize GFLA, the hallunet-complex, H-Nets$_\mathcal{I}$, consists of a texture-to-image flow generator admitting hallucination image poses, $\{\mathbf{p}_i^h\}_{i=1}^{n_h}$, as inputs. Similarly, the texnet-complex, T-Nets$_\mathcal{T}$, consists of a image-to-texture flow generator admitting hallucination poses on both image and texture domain, $\{\mathbf{p}_i^h, \mathbf{b}_i^h\}_{i=1}^{n_h}$, as inputs. Denoting $\phi_\mathcal{I}$ and $\phi_\mathcal{T}$, intermediate features[4] of the image domain and the texture domain, respectively, we can express a generation process of each network complex as

$$\hat{\mathbf{x}}_i^h,\ \phi_{\mathcal{I}i}^h = \text{H-Nets}_\mathcal{I}(\mathbf{x}^s,\ \mathbf{p}^s,\ \mathbf{p}_i^h,\ \phi_\mathcal{T}), \tag{3}$$

$$\mathbf{t},\ \phi_\mathcal{T} = \text{T-Nets}_\mathcal{T}(\mathbf{P}^h,\ \mathbf{B}^h,\ \mathbf{\Phi}_\mathcal{I}^h), \tag{4}$$

where a capital symbol denotes a set of n_h smaller symbols used for hallucinations, *e.g.*, $\mathbf{P}^h = \{\mathbf{p}_i^h\}_{i=1}^{n_h}$ and $\mathbf{\Phi}_\mathcal{I}^h = \{\phi_{\mathcal{I}i}^h\}_{i=1}^{n_h}$.

Hallucination Network Complex

A hallucination network complex consists of a source image encoder, $E_\mathcal{I}^s$, an *image generator*, $G_\mathcal{I}$, and two flow generators: *source-to-target (source-to-hallucination) flow generator*, $F_\mathcal{I}^{st}$, and *texture-to-image flow generator*, $F_{\mathcal{T}\to\mathcal{I}}$.

Image generator. The image generator, $G_\mathcal{I}$, generates n_h pose transferred hallucination images, $\{\hat{\mathbf{x}}_i^h\}_i^{n_h}$, each of which is generated from a source image, $\mathbf{x}^s$, conditioned on a hallucination image pose, $\mathbf{p}_i^h$. The image generator consists of an encoder-decoder structure interleaved with two types of local attention modules – *source local attention module*, $A_\mathcal{I}^{st}$, and *texture local attention module*, $A_{\mathcal{T}\to\mathcal{I}}$, – at decoder side. The source image encoder $E_\mathcal{I}^s$ provides a source image feature extracted from the source image $\mathbf{x}^s$ as a key feature for the source local attention module. Let $\phi_\mathcal{I}^s$ denote a source image feature, $\phi_\mathcal{T}$ denote a texture feature, $\mathbf{f}_\mathcal{I}^{sh}$ and $\mathbf{m}_\mathcal{I}^{sh}$ denote a source-to-hallucination flow and a corresponding mask respectively. Let $\mathbf{f}_{\mathcal{T}\mathcal{I}}$ and $\mathbf{m}_{\mathcal{T}\mathcal{I}}$ denote a texture-to-image flow and a corresponding mask. Then the image generator $G_\mathcal{I}$ generates n_t target images $\{\hat{\mathbf{x}}_i^h\}_{i=1}^{n_h}$ as

$$\hat{\mathbf{x}}_i^h = G_\mathcal{I}(\mathbf{p}_i^h, \phi_\mathcal{I}^s, \phi_\mathcal{T}, \mathbf{f}_\mathcal{I}^{sh}, \mathbf{m}_\mathcal{I}^{sh}, \mathbf{f}_{\mathcal{T}\mathcal{I}}, \mathbf{m}_{\mathcal{T}\mathcal{I}}), \tag{5}$$

referring to the source feature, $\phi_\mathcal{I}^s$, and the texture feature, $\phi_\mathcal{T}$, as a query using a *source-* and a *texture-* local attention module by (1) and (2) respectively.

Source-to-target (source-to-hallucination) flow generator. The source-to-target flow generator, $F_\mathcal{I}^{st}$, generates a source-to-hallucination flow $\mathbf{f}_\mathcal{I}^{sh}$ and a corresponding mask $\mathbf{m}_\mathcal{I}^{sh}$ for the source local attention module from the source image $\mathbf{x}^s$, the source pose $\mathbf{p}^s$ and the hallucination image pose $\mathbf{p}^h$, following the GFLA (Ren et al. 2020), as

$$\mathbf{f}_\mathcal{I}^{sh},\ \mathbf{m}_\mathcal{I}^{sh} = F_\mathcal{I}^{sh}(\mathbf{x}^s,\ \mathbf{p}^s,\ \mathbf{p}^h). \tag{6}$$

Texture-to-image flow generator. The texture-to-image flow generator, $F_{\mathcal{T}\to\mathcal{I}}$, generates a texture-to-image mask, $\mathbf{m}_{\mathcal{T}\mathcal{I}}$, for texture a local attention module basically from a hallucination image pose, $\mathbf{p}^h$, for the lowest layer and sequentially combines a texture feature, $\phi_\mathcal{T}$, of the same level layer after outputting the lowest layer's flow and mask, as depicted in Figure 3.

$$\mathbf{f}_{\mathcal{T}\mathcal{I}},\ \mathbf{m}_{\mathcal{T}\mathcal{I}} = F_{\mathcal{T}\to\mathcal{I}}(\phi_\mathcal{T},\ \mathbf{p}^h). \tag{7}$$

[4]The actual attention mechanism is applied to multiple feature layers, however, we regard them as a single layer feature for a concise representation in the rest of the paper.

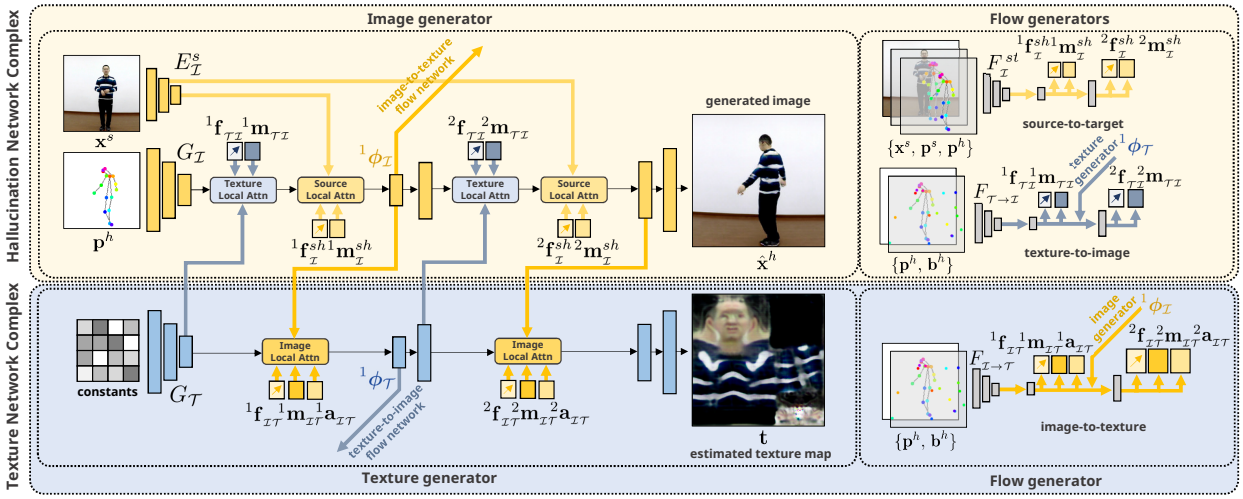

Figure 3: Detailed structures of the hallunet-complex and the texnet-complex consisting of a respective domain generator and the corresponding flow generator(s) with input-output linkages.

Texture Network Complex

A texture network complex consists of a *texture generator*, $G_{\mathcal{T}}$, and a *image-to-texture flow generator*, $F_{\mathcal{I}\to\mathcal{T}}$.

Texture generator. The texture generator, $G_{\mathcal{T}}$, generates an estimated texture map, $\mathbf{t}$, from learn-able constants utilizing multi-pose image features, $\mathbf{\Phi}^t = \{\boldsymbol{\phi}^t\}_{i=1}^{n_t}$. The SMPL (Loper et al. 2015) based texture map we use has a specific layout, in which every body part appears at the same location. Hence, we design the learn-able constants be the universal input for the texture generator, $G_{\mathcal{T}}$, to generate all output texture maps, hoping the texture generator to find an optimal encoding for the universal texture map structure. Similar to the image generator, the texture generator is composed of an encoder-decoder structure interleaved with image local attention modules at decoder side. While the image generator, $G_{\mathcal{I}}$, generates n_t different target images simultaneously, the texture generator, $G_{\mathcal{T}}$, generates a single texture map, $\mathbf{t}$, referring to n_t different image features at once. Let $\mathbf{\Phi}_{\mathcal{I}} = \{\boldsymbol{\phi}_{\mathcal{I}_i}\}_{i=1}^{n_t}$ denote a set n_t image features of n_t different target poses generated by the image generator, and $\mathbf{F}_{\mathcal{IT}} = \{\mathbf{f}_{\mathcal{IT}_i}\}_{i=1}^{n_t}$, $\mathbf{M}_{\mathcal{IT}} = \{\mathbf{m}_{\mathcal{IT}_i}\}_{i=1}^{n_t}$ denote a set of n_t image-to-texture flows and masks, and $\mathbf{a}_{\mathcal{IT}}$ denote an aggregation mask which will be explained later. Then the texture generator, $G_{\mathcal{T}}$, estimates a texture map, $\mathbf{t}$, as

$$\mathbf{t} = G_{\mathcal{T}}(\mathbf{\Phi}_{\mathcal{I}}, \mathbf{F}_{\mathcal{IT}}, \mathbf{M}_{\mathcal{IT}}, \mathbf{a}_{\mathcal{IT}}). \tag{8}$$

We design the texture generator to attend to multi-view features at the same time. To achieve this, we introduce an additional merging layer at the end of the image local attention module to merge multiple multi-view attention features. Let n_h be the number of hallucination and $\boldsymbol{\phi}_i^{attn}$ be the attention feature of i-th view according to (1). Then a merged attention feature, $\boldsymbol{\phi}_{merge}$, is obtained by applying a convolution layer on a concatenation of $\{\boldsymbol{\phi}_i^{attn}\}_{i=1}^{n_h}$ as

$$\boldsymbol{\phi}^{merge} = \text{Conv}(\text{concat}(\boldsymbol{\phi}_1^{attn}, \dots, \boldsymbol{\phi}_{n_h}^{attn})), \tag{9}$$

where concat$(\cdot)$ denotes a concatenation operation on features along the channel dimension. Then the final output feature, $\boldsymbol{\phi}^{out}$, is computed as

$$\boldsymbol{\phi}^{out} = (\mathbf{1} - \mathbf{a}_{\mathcal{IT}}) \otimes \boldsymbol{\phi}_{\mathcal{IT}} + \mathbf{a}_{\mathcal{IT}} \otimes \boldsymbol{\phi}^{merge}, \tag{10}$$

with a texture decoding feature, $\boldsymbol{\phi}_{\mathcal{T}}$, and an aggregation mask, $\mathbf{a}_{\mathcal{IT}}$, obtained by $F_{\mathcal{I}\to\mathcal{T}}$, where $\mathbf{1}$ is a tensor whose elements are all ones.

Image-to-texture flow generator. The image-to-texture flow generator, $F_{\mathcal{I}\to\mathcal{T}}$, generates an image-to-texture flow, $\mathbf{f}_{\mathcal{IT}}$, a corresponding mask, $\mathbf{m}_{\mathcal{IT}}$, and an aggregation mask, $\mathbf{a}_{\mathcal{IT}}$, for image local attention from a hallucination image pose, $\mathbf{p}^h$, a hallucination texture pose, $\mathbf{b}^h$, and a hallucination pixel coordinate, $\mathbf{c}^h$, as

$$\mathbf{f}_{\mathcal{IT}}, \mathbf{m}_{\mathcal{IT}}, \mathbf{a}_{\mathcal{IT}} = F_{\mathcal{I}\to\mathcal{T}}(\mathbf{p}^h, \mathbf{b}^h, \mathbf{c}^h). \tag{11}$$

Notice that the $F_{\mathcal{I}\to\mathcal{T}}$ generates the additional aggregation mask $\mathbf{a}_{\mathcal{IT}}$ for merging multi-view attention features according to (10).

Loss Functions and Training Strategy

We assume that the ground-truth of an estimated texture map is inaccessible. Hence, we render an image using the estimated texture map, $\mathbf{t}$, and compare it to a ground-truth image for training. We simplify the rendering process into warping $\mathbf{t}$ according to a surface coordinate, $\mathbf{s}$. However, the warped texture map constitutes only a foreground human body and lacks background. We provide the lacking background to the rendered image from the generated pose transferred image, $\hat{\mathbf{x}}$, generated by the image generator. Let $\mathbf{m}$ denote a binary mask of the surface coordinate, $\mathbf{s}$. Then we obtain the final rendered image, $\tilde{\mathbf{x}}$, by

$$\tilde{\mathbf{x}} = (\mathbf{1} - \mathbf{m}) \otimes \hat{\mathbf{x}} + \mathbf{m} \otimes warp(\mathbf{t}; \mathbf{s}). \tag{12}$$

Loss functions. To train the image generator, $G_{\mathcal{I}}$, and the texture generator, $G_{\mathcal{T}}$, we use four types of losses in the image domain: Reconstruction loss, $\mathcal{L}_{rec}$, to minimize the difference between a generated/rendered image, $\{\hat{\mathbf{x}}, \tilde{\mathbf{x}}\}$, and ground-truth image, $\mathbf{x}$, according to $\ell 1$ norm as

$$\mathcal{L}_{rec} = \|\hat{\mathbf{x}} - \mathbf{x}\|_1 + \|\tilde{\mathbf{x}} - \mathbf{x}\|_1.$$

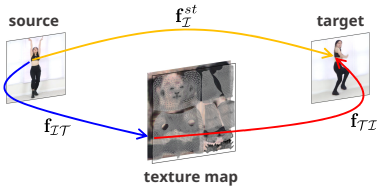

Figure 4: The path along the combined flows, $\mathbf{f}_{\mathcal{IT}}$ and $\mathbf{f}_{\mathcal{TI}}$, should be consistent with a direct path along the $\mathbf{f}_{\mathcal{I}}^{st}$.

The perceptual loss, $\mathcal{L}_{perc}$ (Johnson, Alahi, and Fei-Fei 2016), to minimize $\ell 1$ norm between deep features of $\{\hat{\mathbf{x}}, \tilde{\mathbf{x}}\}$ and $\mathbf{x}$ as

$$\mathcal{L}_{perc} = \sum_{j} \left[\| {}^{j}\boldsymbol{\psi}(\hat{\mathbf{x}}) - {}^{j}\boldsymbol{\psi}(\mathbf{x})\|_1 + \| {}^{j}\boldsymbol{\psi}(\tilde{\mathbf{x}}) - \boldsymbol{\psi}^{j}(\mathbf{x})\|_1 \right],$$

where ${}^{j}\boldsymbol{\psi}$ represents a j-th layer feature obtained by the pre-trained VGG19 networks for preserving coarse level contents. The style loss, $\mathcal{L}_{styl}$ (Johnson, Alahi, and Fei-Fei 2016),

$$\mathcal{L}_{styl} = \sum_{j} \left[\| {}^{j}G_{\psi}(\hat{\mathbf{x}}) - {}^{j}G_{\psi}(\mathbf{x})\|_1 + \| {}^{j}G_{\psi}(\tilde{\mathbf{x}}) - {}^{j}G_{\psi}(\mathbf{x})\|_1 \right],$$

for preserving an overall style, where ${}^{j}G_{\boldsymbol{\psi}}$ represents a Gram matrix constructed from ${}^{j}\boldsymbol{\psi}$. And the hinge version of the adversarial loss, $\mathcal{L}_{adv}$, with a discriminator, $D(\cdot)$, to make the generated/rendered images and the estimated texture map real-looking.

Additionally, we use four types of losses to train three flow generators: $F_{\mathcal{I}}^{st}$, $F_{\mathcal{T}\to\mathcal{I}}$, and $F_{\mathcal{I}\to\mathcal{T}}$. As in Ren et al. (2020), we use the sample correctness loss to train the source-to-target flow generator, $F_{\mathcal{I}}^{st}$:

$$\mathcal{L}_{cor} = \frac{1}{L} \sum_{l\in\Omega} \exp\left(- \frac{\mu(\tilde{\boldsymbol{\psi}}_s^l, \boldsymbol{\psi}_t^l)}{\mu_{max}^l} \right), \tag{13}$$

where $\mu(\cdot,\cdot)$ denotes the cosine similarity, Ω denotes the coordinate set containing all L positions in the feature maps, $\tilde{\boldsymbol{\psi}}_s$ denotes the warping of the VGG19 feature, $\boldsymbol{\psi}_s$, according to the flow $\mathbf{f}_{\mathcal{I}}^{st}$, that is $\tilde{\boldsymbol{\psi}}_s = warp(\boldsymbol{\psi}_s; \mathbf{f}_{\mathcal{I}}^{st})$, with a superscript l denoting feature values of $\tilde{\boldsymbol{\psi}}_s$ located at the coordinate $l = (x, y)$. To train the image-to-texture flow generator, $F_{\mathcal{TI}}$, we introduce a coordinate loss, $\mathcal{L}_{coord}$, as

$$\mathcal{L}_{coord} = \left\| \tilde{\mathbf{m}}_{\mathcal{T}} \otimes (\mathbf{f}_{\mathcal{IT}} - \tilde{\mathbf{c}}) \right\|_2, \tag{14}$$

where $\tilde{\mathbf{c}}$ denotes a rescaled version of $\mathbf{c}$ to the same spatial size and scale of $\mathbf{c}_{\mathcal{IT}}$, $\tilde{\mathbf{m}}_{\mathcal{T}}$ denotes a binary mask indicating visible parts of $\tilde{\mathbf{c}}$. Additionally, we introduce a path consistency loss, $\mathcal{L}_{cons}$. Considering two types of paths as depicted in Figure 4. One path, represented as $\mathbf{f}_{\mathcal{I}}^{st}$, is a direct path from a source to a target. The other path is a two-step path from the source to the target passing through a texture map represented as a combination of image-to-texture flow, $\mathbf{f}_{\mathcal{IT}}$, and texture-to-image flow, $\mathbf{f}_{\mathcal{TI}}$. We assume that information contained in the source image should be convey to the same location on the target image regardless of the paths. To impose this assumption, the path consistency loss reduces the difference between the two paths as

$$\mathcal{L}_{cons} = \left\| \mathbf{m} \otimes \left(\mathbf{f}_{\mathcal{I}}^{st} - warp(\mathbf{f}_{\mathcal{IT}}; \mathbf{f}_{\mathcal{TI}}) \right) \right\|_2, \tag{15}$$

with the binary mask, $\mathbf{m}$, representing foreground human body obtained along with surface annotation, $\mathbf{s}$. Lastly, all flows are regularized by the regularization loss devised in Ren et al. (2020) as

$$\mathcal{L}_{reg} = \mathcal{L}_r(\mathbf{f}_{\mathcal{I}}^{st}) + \mathcal{L}_r(\mathbf{f}_{\mathcal{IT}}). \tag{16}$$

Please refer to Ren et al. (2020) for further details of the regularization loss.

Training Strategy The goal of the hallunet-complex is to provide sufficient image features from diverse viewpoints to the texture generator. Providing evenly rotated poses as a set of hallucination poses could be an option, however, generating evenly rotated images is often ungeneralizable for the image generator as most training images are biased to frontal and side views. To balance the trade-off between viewpoint diversity and generalization performance, we sample n_h−1 poses from another image pair having a different clothes identity and combine a source pose to make a set of n_h hallucination poses. Let $\{\mathbf{p}_i^h\}_{i=1}^{n_h}$ denote a set of hallucination image poses. As $\{\mathbf{p}_i^h\}_{i=1}^{n_h}$ are sampled from the other image pair, except one from the source, we do not have ground-truths to evaluate the generated hallucination images. Hence we propose two-step generation processes for training as depicted in Figure 2. Firstly, we run the whole networks, both hallunet-complex and texnet-complex, using the sampled hallucination poses, $\{\mathbf{p}_i^h\}_{i=1}^{n_h}$, to obtain an estimated texture map, $\mathbf{t}$, and a texture feature, $\boldsymbol{\phi}_{\mathcal{T}}$, which we named it $\boldsymbol{\phi}_{\mathcal{T}}^{hallu}$. Let $\{\mathbf{p}_i^t\}_{i=1}^{n_t}$ denote target image poses of the current image pair, a set of different pose images of the source image which we can use as ground-truths. In the second step, we run the hallunet-complex solely using the target image poses, $\{\mathbf{p}_i^t\}_{i=1}^{n_t}$, referring to the kept hallucination texture feature, $\boldsymbol{\phi}_{\mathcal{T}}^{hallu}$, to obtain generated target images, $\{\hat{\mathbf{x}}_i^t\}_{i=1}^{n_t}$, posing $\{\mathbf{p}_i^t\}_{i=1}^{n_t}$. Now, we do have the ground-truths for $\{\hat{\mathbf{x}}_i^t\}_{i=1}^{n_t}$, we can train the whole networks using the proposed loss functions.

Experiments

Datasets. We use three datasets to evaluate our model: DeepFahsion In-shop Clothes Retrieval Benchmark (Liu et al. 2016), iPER (Liu et al. 2019), and Fashion video collected from Amazon (Zablotskaia et al. 2019). From DeepFahsion we filter out 5,745 images wearing 1,628 different clothes, which are non-detectable to the human detector (Cao et al. 2019), from the training set.

Evaluation and metrics. To evaluate estimated texture maps, we render multi-pose/view images using the estimated texture maps as we do not have ground-truth texture maps. We use three measures to compute reconstruction errors and a distributional discrepancy between generated images and reference images: Structural similarity (SSIM)

Figure 5: Examples of comparison result on Deepfashion dataset to pose transferred image generation methods. LWG (Liu et al. 2019) and GFLA (Ren et al. 2020) preserve textures locally, but sometimes fail to generate exact posture and scaled images, while ours preserve the overall postures and scales.

	DeepFashion			iPER	
	FID ↓	LPIPS ↓	SSIM ↑	LPIPS ↓	SSIM ↑
PG2	47.714	0.246	0.763	0.135	0.854
Def-GAN	18.457	0.233	0.761	0.129	0.829
LWG	23.286	0.283	0.731	0.087	0.840
GFLA	10.573	0.234	0.715	-	-
HPBTT	-	-	0.735	-	-
Ours ($G_{\mathcal{I}}$)	**9.001**	**0.156**	**0.830**	**0.051**	**0.907**
Ours ($G_{\mathcal{T}}$)	19.656	0.177	0.786	0.063	0.897

Table 1. Comparison results on DeepFashion and iPER datasets. $G_{\mathcal{I}}$ indicates the image domain output and $G_{\mathcal{T}}$ indicates the rendered image using the estimated texture map.

(Wang et al. 2004), Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al. 2018), and Fréchet inception distance (FID) (Heusel et al. 2017).

Comparisons

We compare our method to several state-of-the-art pose guided image transfer methods including PG2 (Ma et al. 2017), Def-GAN (Siarohin et al. 2018), GFLA (Ren et al. 2020), LWG (Liu et al. 2019), and a recent texture map estimation method, HPBTT (Zhao et al. 2020). The results are summarized in Table 1. On the DeepFashion and iPER dataset our method outperforms the others. Figure 5 shows some examples comparing generated images (ours ($G_{\mathcal{I}}$)) and rendered images (ours ($G_{\mathcal{T}}$)) of our method to other pose transferred image generation methods. We can find that our model has the advantage of generating an image involving large scale transform over the others. For example, in the third row of Figure 5, LWG and GFLA generate fine details locally but fail at generating the exact pose and consistent patterns overall. However, our method successfully generates a desirable scaled image even if the target pose

	iPER		FashionVid	
	LPIPS ↓	SSIM ↑	LPIPS ↓	SSIM ↑
$G_{\mathcal{I}}$ ($n_h = 1$)	0.067	0.894	0.063	0.920
$G_{\mathcal{I}}$ ($n_h = 2$)	0.055	**0.907**	**0.061**	0.922
$G_{\mathcal{I}}$ ($n_h = 3$)	**0.051**	**0.907**	0.066	0.921
$G_{\mathcal{I}}$ ($n_h = 4$)	0.082	0.871	**0.061**	**0.924**
$G_{\mathcal{T}}$ ($n_h = 1$)	0.076	0.887	0.081	0.903
$G_{\mathcal{T}}$ ($n_h = 2$)	0.066	**0.898**	0.081	**0.905**
$G_{\mathcal{T}}$ ($n_h = 3$)	0.063	0.897	0.082	0.903
$G_{\mathcal{T}}$ ($n_h = 4$)	**0.026**	0.869	**0.079**	**0.905**

Table 2. Analysis results of the number of hallucination.

	iPER		FashionVid	
	LPIPS ↓	SSIM ↑	LPIPS ↓	SSIM ↑
$G_{\mathcal{I}}$ (*Ind.*)	**0.051**	**0.907**	0.073	0.913
$G_{\mathcal{I}}$ ($\mathcal{T}\rightarrow\mathcal{I}$)	0.056	0.903	0.072	0.916
$G_{\mathcal{I}}$ ($\mathcal{I}\rightarrow\mathcal{T}$)	0.061	0.901	**0.057**	0.900
$G_{\mathcal{I}}$ (*Full*)	**0.051**	**0.907**	0.066	**0.921**
$G_{\mathcal{T}}$ (*Ind.*)	0.075	0.888	0.100	0.888
$G_{\mathcal{T}}$ ($\mathcal{T}\rightarrow\mathcal{I}$)	0.075	0.887	0.085	0.901
$G_{\mathcal{T}}$ ($\mathcal{I}\rightarrow\mathcal{T}$)	0.073	0.892	0.083	0.902
$G_{\mathcal{T}}$ (*Full*)	**0.063**	**0.897**	**0.082**	**0.903**

Table 3. Results of the ablation study. $G_{\mathcal{I}}$ denotes generated image from the image generator and $G_{\mathcal{T}}$ denotes rendering image using the estimated texture map.

represents merely a magnified body part, ascribing to the interacting feature flow. The rendering results of ours are comparable to the others despite some artifacts, attributing to resolution mismatch between the image and the texture map and surface annotation errors. Thus, we can conclude that our texture generator generates a plausible texture map for direct rendering. Please refer to the supplementary material for more examples.

Analysis and Ablation Study

Number of hallucination. To analyze whether the proposed hallucination generation scheme is indeed helpful for texture map estimation, we conduct experiments increasing the number of hallucination, n_h, from one to four. Table 2 summarizes the results. For iPER, LPIPS tends to decrease for increasing n_h on both generated and rendered images. In terms of SSIM, generation quality greatly increases for n_h=2 compared to n_h=1 for all cases, which demonstrates the effectiveness of the proposed hallucination generation scheme for texture map estimation. However, there are little improvements for $n_h > 2$ and a degenerate result appears for $G_{\mathcal{T}}(n_h$=4). Practically, as each posed image reveals a half of the whole surface, n_h=2 seems sufficiently enough to contain all surface features. We conjecture the degenerate result for n_h=4 on iPER is attributed to overlapping surfaces among hallucinations which distract both image and texture generators from generating qualified outputs.

Ablation study. To analyze the role of inter-domain feature flows, we conduct ablation studies by unlinking each attention path from one domain to the other. *Independent model (Ind.)* has no inter-domain attention path, *Image-*

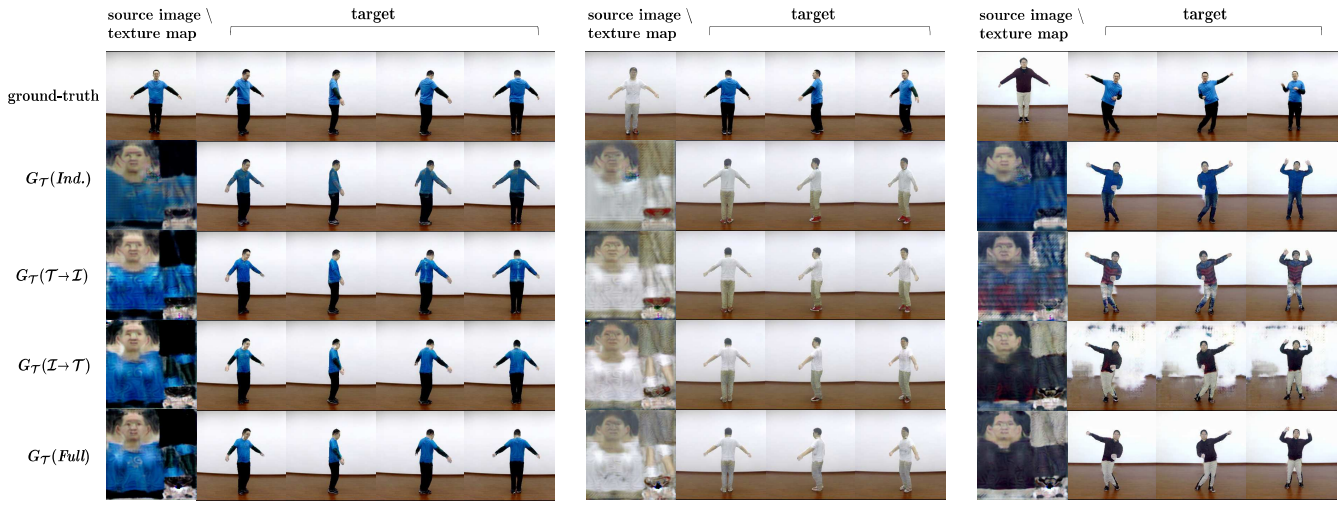

Figure 6: Examples of ablation studies. Rendered images of various viewpoints and poses using estimated texture maps from source images (column 1 of each example, top: source image, rest: texture map).

to-texture model ($\mathcal{I}\rightarrow\mathcal{T}$) solely has image-to-texture attention path, and *Texture-to-image models ($\mathcal{T}\rightarrow\mathcal{I}$)* solely has texture-to-image attention path. As the original input for the texture generator have no distinguishable information about input clothes, we leave the image-to-texture flow at the lowest layer solely from the source image to the texture generator for the *Ind.* and the $\mathcal{T}\rightarrow\mathcal{I}$ models. *Full model (Full)* denote the original model consisting of all directional attention paths and $n_h=3$ is used. The results are summarized in Table 3. The quality of estimated texture maps ($G_{\mathcal{T}}$), evaluated by rendered images, improves when image-to-texture flows are added to the independent model, and improves further when texture-to-image flows are incorporated. For the image generator ($G_{\mathcal{I}}$), neither the $\mathcal{I}\rightarrow\mathcal{T}$ nor $\mathcal{T}\rightarrow\mathcal{I}$ model shows a consistent improvement, however, the two types of flow altogether improve the overall generation quality. Figure 6 shows some rendered images comparing ablated models. In Figure 6, the *Ind.* and the $\mathcal{T}\rightarrow\mathcal{I}$ models cannot generate distinguishable black sleeves in the first example while $\mathcal{I}\rightarrow\mathcal{T}$ model and the full model generate distinguishable black sleeves. The *Ind.* and $\mathcal{T}\rightarrow\mathcal{I}$ models often fail at generating accurate clothes color while the full model succeed. The $\mathcal{I}\rightarrow\mathcal{T}$ model generates comparable texture map to the full model, however, the it often generates some artifacts on texture map and background.

Application

To verify the usability of our method for 3D model rendering, we generate 3D animation clips using texture maps generated by our method. We first reconstruct a sequence of 3D human shape in the SMPL format using the off-the-shelf 3D video reconstruction model (Kocabas, Athanasiou, and Black 2020) and the off-the-shelf clothing model (Ma et al. 2020). Then we apply a texture map generated by ours to the 3D shape sequence to obtain a colored animation clip. Figure 7 shows some examples. The generated 3D animations are viewed in two different viewpoints. The result shows that

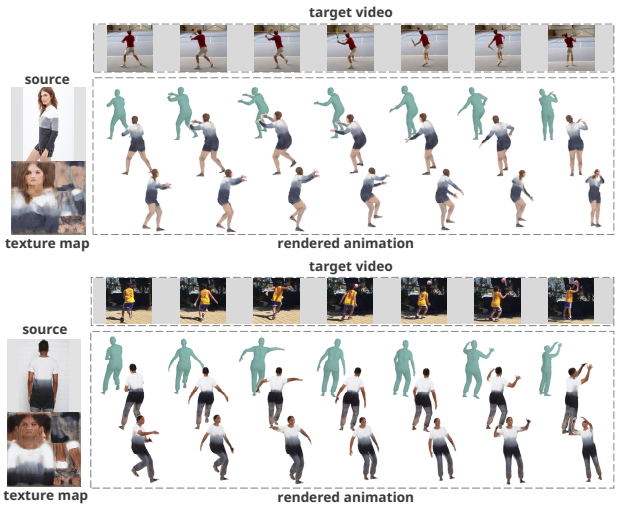

Figure 7: Examples of 3D animation rendering using our estimated texture map viewed in two different viewpoints.

a person in a source image acts as the target target video, preserving clothes patterns all around. The generated texture map works effectively for 3D model rendering, generating consistent images for any pose and viewpoint.

Conclusion

We propose dual-domain generative models for a complete texture map estimation by providing multi-view features using a novel hallucination generation scheme. Our model utilizes a local attention module over the domains to convey multi-view features to the texture map and texture features to pose transferred images. Experimental results show that the estimated texture map has decent quality for rendering colorful 3D human models, which is applicable to generate a free-view point 3D animation.

Acknowledgments

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-01190, [SW Star Lab] Robot Learning: Efficient, Safe, and Socially-Acceptable Machine Learning).

References

Alldieck, T.; Magnor, M.; Bhatnagar, B. L.; Theobalt, C.; and Pons-Moll, G. 2019. Learning to Reconstruct People in Clothing From a Single RGB Camera. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Bogo, F.; Kanazawa, A.; Lassner, C.; Gehler, P.; Romero, J.; and Black, M. J. 2016. Keep It SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In *Eur. Conf. Comput. Vis.* Springer.

Cao, Z.; Hidalgo, G.; Simon, T.; Wei, S.-E.; and Sheikh, Y. 2019. OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(1): 172–186.

Catmull, E. 1974. *A Subdivision Algorithm for Computer Display of Curved Surfaces*. PhD dissertation, Utah Univ Salt Lake City School of Computing.

Catmull, E.; and Smith, A. R. 1980. 3-D Transformations of Images in Scanline Order. In *SIGGRAPH*, 279–285. ACM.

Choutas, V.; Pavlakos, G.; Bolkart, T.; Tzionas, D.; and Black, M. J. 2020. Monocular Expressive Body Regression Through Body-Driven Attention. In *Eur. Conf. Comput. Vis.* Springer.

Gabeur, V.; Franco, J.-S.; Martin, X.; Schmid, C.; and Rogez, G. 2019. Moulding Humans: Non-Parametric 3D Human Shape Estimation From Single Images. In *Int. Conf. Comput. Vis.*

Grigorev, A.; Sevastopolsky, A.; Vakhitov, A.; and Lempitsky, V. 2019. Coordinate-Based Texture Inpainting for Pose-Guided Human Image Generation.

Güler, R. A.; Neverova, N.; and Kokkinos, I. 2018. DensePose: Dense Human Pose Estimation in the Wild. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Adv. Neural Inform. Process. Syst.*

Jian, W.; Yunshan, Z.; Yachun, L.; Chi, Z.; and Yichen, W. 2019. Re-Identification Supervised Texture Generation. *IEEE Conf. Comput. Vis. Pattern Recog.*

Johnson, J.; Alahi, A.; and Fei-Fei, L. 2016. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In *Eur. Conf. Comput. Vis.* Springer.

Kanazawa, A.; Black, M. J.; Jacobs, D. W.; and Malik, J. 2018. End-to-End Recovery of Human Shape and Pose. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Kocabas, M.; Athanasiou, N.; and Black, M. J. 2020. VIBE: Video Inference for Human Body Pose and Shape Estimation. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Kolotouros, N.; Pavlakos, G.; Black, M. J.; and Daniilidis, K. 2019. Learning to Reconstruct 3D Human Pose and Shape via Model-Fitting in the Loop. In *Int. Conf. Comput. Vis.*

Lassner, C.; Romero, J.; Kiefel, M.; Bogo, F.; Black, M. J.; and Gehler, P. V. 2017. Unite the People: Closing the Loop Between 3D and 2D Human Representations. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Lazova, V.; Insafutdinov, E.; and Pons-Moll, G. 2019. 360-Degree Textures of People in Clothing from a Single Image. In *Int. Conf. 3D Vision (3DV)*. IEEE.

Liu, W.; Piao, Z.; Min, J.; Luo, W.; Ma, L.; and Gao, S. 2019. Liquid Warping GAN: A Unified Framework for Human Motion Imitation, Appearance Transfer and Novel View Synthesis. In *Int. Conf. Comput. Vis.*

Liu, Z.; Luo, P.; Qiu, S.; Wang, X.; and Tang, X. 2016. DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Loper, M.; Mahmood, N.; Romero, J.; Pons-Moll, G.; and Black, M. J. 2015. SMPL: A Skinned Multi-Person Linear Model. *ACM Trans. Graph.*, 34(6): 248:1–248:16.

Ma, L.; Jia, X.; Sun, Q.; Schiele, B.; Tuytelaars, T.; and Van Gool, L. 2017. Pose Guided Person Image Generation. In *Adv. Neural Inform. Process. Syst.*

Ma, Q.; Yang, J.; Ranjan, A.; Pujades, S.; Pons-Moll, G.; Tang, S.; and Black, M. J. 2020. Learning to Dress 3D People in Generative Clothing. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Natsume, R.; Saito, S.; Huang, Z.; Chen, W.; Ma, C.; Li, H.; and Morishima, S. 2019. SiCloPe: Silhouette-Based Clothed People. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Neverova, N.; Alp Guler, R.; and Kokkinos, I. 2018. Dense Pose Transfer. In *Eur. Conf. Comput. Vis.*

Pavlakos, G.; Zhu, L.; Zhou, X.; and Daniilidis, K. 2018. Learning to Estimate 3D Human Pose and Shape From a Single Color Image. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Ren, Y.; Yu, X.; Chen, J.; Li, T. H.; and Li, G. 2020. Deep Image Spatial Transformation for Person Image Generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Saito, S.; Huang, Z.; Natsume, R.; Morishima, S.; Kanazawa, A.; and Li, H. 2019. PIFu: Pixel-Aligned Implicit Function for High-Resolution Clothed Human Digitization. In *Int. Conf. Comput. Vis.*

Saito, S.; Simon, T.; Saragih, J.; and Joo, H. 2020. PIFuHD: Multi-Level Pixel-Aligned Implicit Function for High-Resolution 3D Human Digitization. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Siarohin, A.; Sangineto, E.; Lathuilière, S.; and Sebe, N. 2018. Deformable GANs for Pose-Based Human Image Generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Thormählen, T.; and Seidel, H.-P. 2008. 3D-Modeling by Ortho-Image Generation from Image Sequences. *ACM Trans. Graph.*, 27(3): 1–5.

Varol, G.; Ceylan, D.; Russell, B.; Yang, J.; Yumer, E.; Laptev, I.; and Schmid, C. 2018. BodyNet: Volumetric Inference of 3D Human Body Shapes. In *Eur. Conf. Comput. Vis.* Springer.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *Adv. Neural Inform. Process. Syst.*

Wang, L.; Kang, S. B.; Szeliski, R.; and Shum, H.-Y. 2001. Optimal Texture Map Reconstruction from Multiple Views. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. Image Process.*, 13(4): 600–612.

Weng, C.-Y.; Curless, B.; and Kemelmacher-Shlizerman, I. 2019. Photo Wake-Up: 3D Character Animation From a Single Photo. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Zablotskaia, P.; Siarohin, A.; Zhao, B.; and Sigal, L. 2019. DwNet: Dense warp-based network for pose-guided human video generation. In *Brit. Mach. Vis. Conf.*

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *IEEE Conf. Comput. Vis. Pattern Recog.*

Zhao, F.; Liao, S.; Zhang, K.; and Shao, L. 2020. Human Parsing Based Texture Transfer from Single Image to 3D Human via Cross-View Consistency. In *Adv. Neural Inform. Process. Syst.*

Zhi, T.; Lassner, C.; Tung, T.; Stoll, C.; Narasimhan, S. G.; and Vo, M. 2020. TexMesh: Reconstructing Detailed Human Texture and Geometry from RGB-D Video. In *Eur. Conf. Comput. Vis.* Springer.