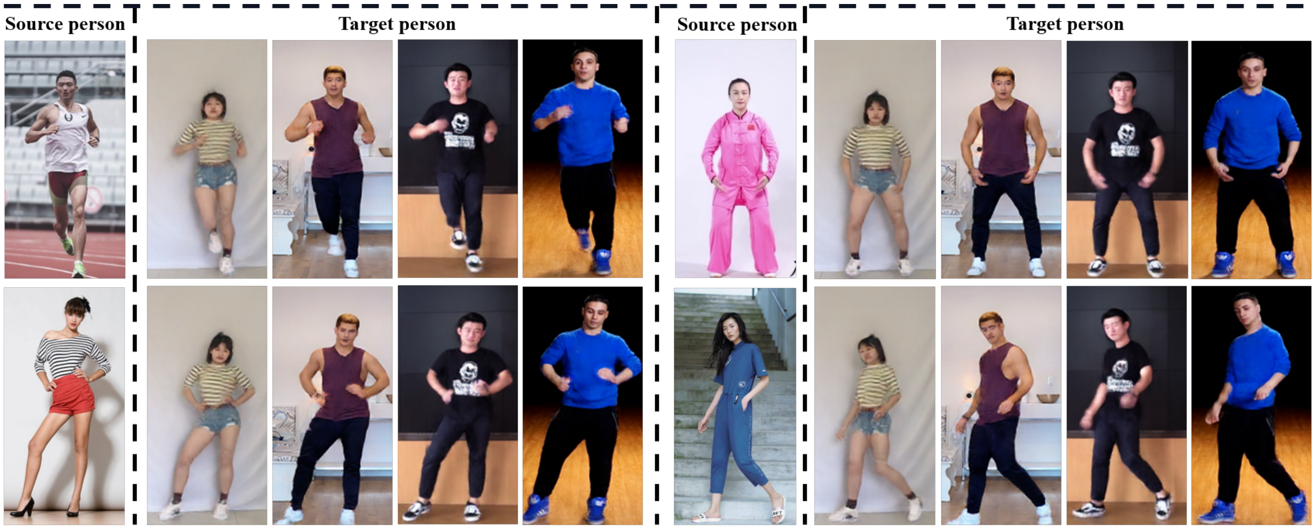
# BPA-GAN: Human Motion Transfer Using Body-Part-Aware Generative Adversarial Networks

Anonymous cvm submission

Paper ID 279



Our BPA-GAN transfers the motion of the source person to a target person. The visual components of the target images, such as the background, the body shape, the facial expressions are successfully preserved after the motion retargeting.

## Abstract

**Human motion transfer has many applications in human behavior analysis, training data augmentation, and personalization in mixed reality. We propose a Body-Parts-Aware Generative Adversarial Network (BPA-GAN) for image-based human motion transfer. Our key idea is to take advantage of the human body with segmented parts instead of using the human skeleton like most of existing methods to encode the human motion information. As a result, we improve the reconstruction quality, the training efficiency, and the temporal consistency via training multiple GANs in a local-to-global manner and adding regularization on the source motion. Extensive experiments show that our method outperforms the baseline and the state-of-the-art techniques in preserving the details of body parts.**

## 1. Introduction

Human motion transfer is crucial in creating human-centric visual content such as making animations about a human acting complex motions, synthesizing training data for learning autonomous driving, personalizing an avatar in virtual reality, etc. On the other hand, as it is expensive and tedious to produce a large amount of human motion data, transferring the motion of a source video to any target person is of great significance for computer graphics and vision communities [8, 1].

Human motion transfer aims to migrate the motion from a source image (video) to a target image (video) while preserving both the background and the the visual appearance of the target person. The main technical challenges include (1) extracting the motion information from the source, (2) encoding the visual information of the target, and (3) synthesizing a new image that combines the above motion and visual information.

Deep neural networks have shown its superiority in high-level image manipulation tasks. A lot of methods [8, 1, 24] correspondingly adopt a three-component framework to address the issue. First, the pose information from the source and target images are extracted. Then, a Conditional Generative Adversarial Network (CGAN) [19] is employed to learn a latent space that encodes the visual information of the target images. Finally, the trained model predicts the motion-retargeted images by using source poses as the condition. Though these methods have achieved stunning motion transfer results, they usually require training on a large dataset in order to recover crisp details of the human body parts.

This paper introduces a body-part-aware generative adversarial network (BPA-GAN) that improves the aforementioned human motion transfer framework by considering the body part information. The improvement includes three aspect: (1) instead of extracting a simple skeleton representation from images, we segment the 2D human body into a map of rigid parts that provide valuable conditions, such as the shapes of and the boundaries between different visual components; (2) rather than encoding the entire image at once, we train GANs that first predict the body parts locally before combining them together globally; (3) we improve the temporal consistency of the output video by regularizing the motion extracted from the source. As a result, our BPA-GAN allows reconstructing the target person's fine-grained details using only a handful of target images.

The core of our method is detecting and segmenting human body parts in RGB images, which are actually an ill-posed problem due to the ambiguity in human body shape, dressing, occlusion, and lighting condition [5, 3, 17]. Our main idea is to leverage the power of data-driven human body fitting methods [28, 29, 30] to estimate a full 3D body model from the images, and then project the 3D model onto these images to guide their segmentation.

We perform extensive experiments and show that our method outperforms the state-of-the-art methods in both detail preservation and training efficiency. Our ablation study justifies that our network design is appropriate. The code, the dataset, and the trained models will be publicly available for future research.

## 2. Related Work

Most of the deep learning based human motion transfer methods train a Generative Adversarial Network (GAN) [13] to generate new poses of the target person. These approaches can be roughly put into two categories according to their generalization ability. The first category is based on a generic model and designed to transfer poses of sources to arbitrary targets [4, 36, 24, 40] without having to retrain or fine-tune the model for an unseen target. The second category is based on a personalized model, which focuses on learning the appearance of a specific person and only generates new poses of the same person [8, 38, 1, 23, 33].

### 2.1. Methods based on generic models

Generic methods are generally trained on a large dataset consisting of images/videos from a couple of people. They usually suffer difficulty in preserving fine details due to lack of information on the target person's appearance.

Balakrishnan et al. [4] segment a scene into the background and a set of body parts by using UNet [31] and warp the body parts into final results. Neverova et al. [27] employ DensePose [2] to map pixels of images onto a common surface and design a blending module to mix the predicted images and warped images. Zhao et al. [45] propose a framework consisting of three networks, a motion condition network for predicting the future poses, a motion forecasting network for transferring the current pose, and a motion refinement network for smoothing the generated sequence. Bellini et al. [6] introduce a video-based post-processing method that maximizes the number of suitable matches between motions and music beats.

Cheng et al. [10] extract background features and six residual network blocks [15] and a self-attention block [43] to merge the information from multiple perspectives. Wang et al. [35] synthesize an image which have the same style with the style image. To preserve the consistency of the style, they use a style consistency discriminator, an adaptive semantic consistency loss and trained the model with a data sampling strategy. Chen et al. [9] take a local-to-global approach to synthesize the face images from sketches. They first learn the feature embeddings of key face components and then map the embedded features to realistic images. Wang et al. [36] dynamically configure the video synthesis through the network weight generation to generalize the network to other target persons.

Wei et al. [40] exploit the body part information to achieve an appearance-controllable video motion transfer, where an instance-level human parsing network [12] is used to extract body pasts semantic layouts from an input frame.

Noting that the aforementioned methods only exploit the 2D information during transferring, some researchers try to utilize 3D information. Guan et al. [14] employ HMR [21] to fit the image with human body parametric model SMPL [25] and combine the corresponding relationship between images pixels and feature points provided by DensePose [2] to add texture to the SMPL mesh model and generate the final result through VUnet [11]. Liu et al. [24] first fit the human body model by HMR and interpolate the images according to the model vertices to obtain rough results. Then they refine the rough results by Liquid Warping GAN. Inspired by these works, our method employs the SMPLify-X [29] model to extract the 3D information from the input images.

### 2.2. Methods based on personalized models

For better detail preservation, personalized models are only trained with one target person. These kind of methods usually require more training data about the target person.

Isola et al [19] propose pix2pix using condition generators and patch discriminators for image-to-image translation. Wang et al. [38] improve pix2pix by proposing pix2pixHD for generating high-resolution images with a coarse-to-fine generator and a multi-scale discriminator. Zhou et al. [46] use STN [20] to transfer texture of the target person onto the the body parts and then fuse the parts together. These methods get good results in images, while some methods try to generate temporally coherent videos. Chan et al. [8] make use of two adjacent frames to train pix2pixHD to improve the temporal coherence and employ an additional face GAN to improve facial details. Wang et al. [37] propose vid2vid for video-to-video translation based on pix2pixHD with the optical flow loss extracted by Flownet2 [18]. Aberman et al. [1] take advantage of paired data to train the network and employ the optical flow between unpaired data to improve the temporal coherence.

Some methods exploit 3D space information to improve results. Liu et al. [23] reconstruct a textured human mesh and combine the mesh with a human parametric model [42]. Then, they render the model to get intermediate results and use a re-weighted PatchGAN [19] to refine these results. Sun et al. [33] leverage the approach in [41] to get human models, and input the projection images of models as well as appearance images into MT-Net and DE-Net for generating and refining the outputs. We refer to a survey by Wang et al. [34] for relevant works that use deep learning methods for VR content creation and exploration.

Our method is also based on personalized models, but we focus on exploiting the body-parts prior knowledge to improve the training efficiency and the quality of results.

## 3. Method

As having been mentioned, our BPA-GAN is a personalized approach. Given a set of images of a target person, we train a model to learn a latent space that encodes the visual information of these images. In the inference stage, given a source image of arbitrary people, BPA-GAN generates a new image of the target person but with the same pose as the source human. Instead of using the conventional skeleton representation, we view the human pose as a 2D human segmented into 16 body parts. Therefore, our method requires a preprocessing step to generate body-part maps from given images. Compared with 2D skeletons, the marks of body-parts contain more information about the body shape and the occlusion relationship between body parts. It also allows us to add additional supervision to body part levels.

### 3.1. Body-part segmentation

Parsing the human body into different parts plays a crucial role in our method, which is rather challenging since the body region usually occupies a small area in an image. We propose a segmentation approach based on a 3D geometric agent.

As shown in Figure 1, it first employs SMPL-X [29], a human parametric representation, to fit body shape and pose, facial expression, hand gesture as well as camera orientation from the input image. This process is also known as SMPLify-X in [29]. We render the reconstructed 3D model to obtain a pose image, in which each body part is in unique pseudo color. The body region of the input image can then be segmented into 16 parts according to pixel color labels of the pose image.

**Human model fitting.** SMPL-X is a human body parametric model that consists of an average human body mesh template $M$ and three sets of control parameters. Template $M$ includes 10475 vertices and $K = 54$ joints. The three sets of parameters are respectively shape parameters $\beta$ $in\ \mathbb{R}^{10}$, pose parameters $\theta \in \mathbb{R}^{3(K+1)}$, and facial expression parameters $\psi \in \mathbb{R}^{10}$. The parametric model can be denoted as $M(\beta, \theta, \phi)$. It indicates that we can use it to evaluate the new positions of mesh vertices for arbitrary given parameters.

It is often required to evaluate the above shape and motion parameters for a given 3D or 2D shape and pose information as an inverse problem. We directly employ SMPLify-X in [29] to estimate parameters $\beta, \theta, \phi$ of SMPL-X and camera parameters $c \in \mathbb{R}^3$ from a given image. In SMPLify-X, fitting SMPL-X to the image is cast into minimizing the following energy [29]:

$$\begin{aligned} E\left(\{\beta\},\{\theta\},\{\varphi\}\right) = E_J + \lambda_{\theta_b} E_{\theta_b} + \lambda_{\theta_f} E_{\theta_f} + \lambda_{\theta_h} E_{\theta_h} \\ + \lambda_\alpha E_\alpha + \lambda_\beta E_\beta + \lambda_\psi E_\psi + \lambda_C E_C \end{aligned} \tag{1}$$

where the first item in the righthand side is losses of 2D joints; items 2-5,7 are respectively priors for body pose, facial pose, hand pose, elbow and knee bending, and facial expression; item 6 penalizes the deviation of shape parameters from the distribution.

**Human model fitting for videos.** SMPLify-X is designed for dealing with a single image. Using it to independently fit SMPL-X to each frame in a video usually leads to temporally inconsistent and flickering output. This is because (1) the shape parameters are not fixed for the same video (see the blue curve in the top of Figure 10 for example), and (2) pose parameters fail to transition between adjacent frames smoothly (see the blue curve in the bottom of Figure 10. To address the issues and simultaneously sustain simplicity, we still employ the frame-by-frame fitting strategy, but add temporal constraints on the current frame parameters using the estimated values of its two previous ones.

Specifically, we employ the original SMPLify-X to fit the first frame of the video. Let $\beta_1$ be the recovered shape parameters. $E_\beta\left(\beta\right)$ in Equation 1 The following soft constraint which forces the shape parameters $\beta$ of the current frame changing as small as possible:

$$E_\beta\left(\beta\right) = \left\|\beta - \beta_1\right\|^2 \tag{2}$$

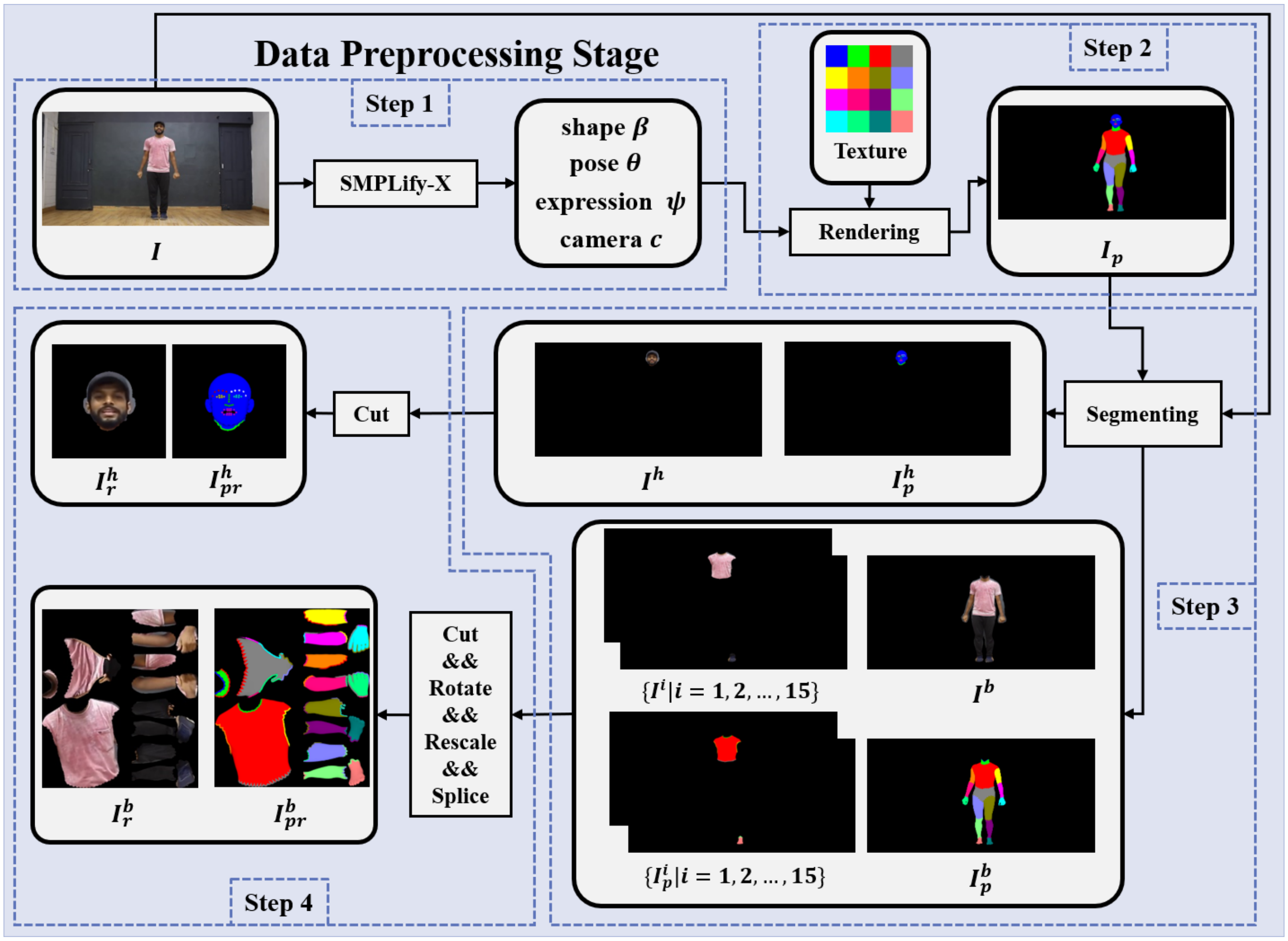


Figure 1. Pipeline of human body part segmentation: (1) SMPL-X fitting; (2) 3D model rendering with pseudo colors $I_p$; (3) Segmentation of body region; (4) Body part layout.

As for pose and expression parameters, we employ the Laplacian operator to constrain the parameters of the current frame. With parameters of its two previous frames given, denoted by $\theta^x_{-2}, \theta^x_{-1}, x \in \{'b',' f',' h'\}$ and $\psi_{-2}, \psi_{-1}$ respectively, we use the following energy to replace the corresponding energy in Equation 1

$$\begin{aligned} E_{\theta_x} &= \left\|\theta^x_{-2} - 2\theta^x_{-1} + \theta^x\right\|^2, x \in \{'b',' f',' h'\} \\ E_{\psi} &= \|\psi_{-2} - 2\psi_{-1} + \psi\|^2 \end{aligned} \tag{3}$$

**Human model rendering and body part segmentation.** As a preprocessing step, we assume the mesh template $M$ of SMPL-X has been segmented into 16 body parts, which are labeled by using 16 selected colors. We use extra six selected colors to label feature points of eyebrows, eyes, nose, and mouth for further capturing facial details. With these specified colors, we can visualize a pseudo color image, including these body parts and feature points, as shown in Figure 2 in which a standard pose is depicted, and subsequently, all body parts and facial feature points are visible. For different shapes, poses, and views, the rendered images will look completely different.

To parse the human body in real image $I$, we employ the above improved SMPLify-X to recover the 3D human model, i.e., estimating the shape, pose, and expression parameters of SMPL-X, as well as camera parameters. Then we segment the human body in $I$ to 16 parts to obtain pseudo color image $I_p$ with six types of feature points on the face part, which we call the pose map. Note that, at the training stage, both shape and pose parameters of the SMPL-X model are from the target person. However, at inference stage, we combine the shape parameters from the target person, the pose and expression parameters of the source person, and the scaled camera parameters of the source person according to the depth to generate the segmented pose. This guarantees that the body scale of the inference input is accordant with that of the training inputs.

**Body part re-layout: dataset generation.** We gather training data from Internet videos. The original images have

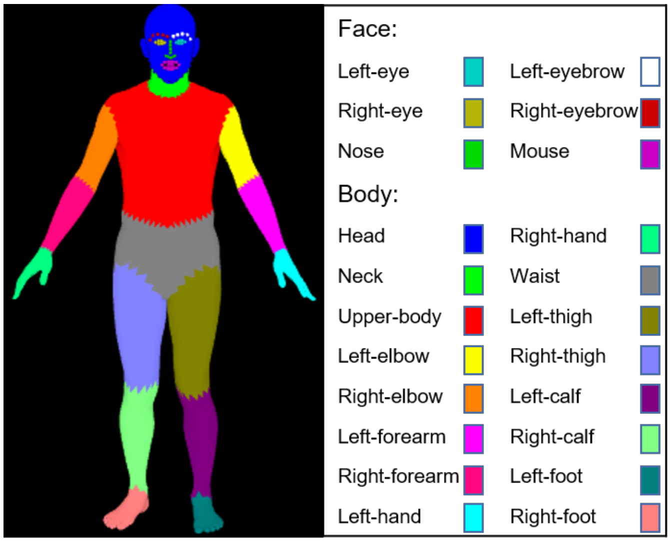


Figure 2. Different body parts are corresponding to different colors of the pose images.

1920×1080 pixels, and we rescale them to 256×256 images as input. Directly resizing the original images to $512 \times 256$ resolution will make small body parts such as the face losing detailed information seriously.

The face is an essential part of human appearance; it is vital for a pose transfer algorithm to a composite clear and real face. We segment the head region out and up-scale it before saving it in an independent image.

As shown in Figure 1, we first segment out head image $I^h$ from $I$ and the corresponding head pose image $I^h_p$, and then cut the head region to obtain the resized head image $I^h_r$ and the resized head pose map $I^h_{pr}$, both of which are of $256 \times 256$.

Similarly, the rest 15 body parts are also segmented out to obtain part images, and the corresponding pose maps $I^i, I^i_p, i = 1, 2, \cdots, 15$. We use the bounding boxes to cut each part and calculate the angle to rotate these bounding boxes to horizontal or vertical. The orientations depend on different parts. Then, we use these angle to rotate the part images $I^i$ and the corresponding pose maps $I^i_p$. In the final layout, $256 \times 256$ pixels are for the upper body, $256 \times 192$ pixels are for the waist, and $128 \times 64$ pixels are for each rest 13 body parts. So, we rescale these parts by the smaller scaling ratios between the width and height. We then use zero paddings to fill the images to the size we design. We splice these parts to get a body part image $I^b_r$ and the corresponding pose map $I^b_{pr}$. Besides, we record the positions and rotation angle of all the human parts in the original images in order to reposition them, as shown in the mid-output of Figure 1. Finally, each of the training images yields a texture image and a pose map for the head, and a texture image and a pose map for other parts.

### 3.2. BPA-GAN

Unlike most existing approaches that generate the entire image containing humans, BPA-GAN only predicts human body parts and then combine them into a full human body. Particularly, it contains three modules as shown in Figure 3: the part generation module estimates the head image $\hat{I}^h_r$ and the body part images $\hat{I}^b_r$ from the input pose maps; the intermediate model recovers body parts to their original size, and the fusion module stitches the body parts as well as the given background together to yield a full image. $\hat{I}$.

**Generators.** BPA-GAN includes three generators, as shown in Figure 3. The two generators of the part generation module are responsible for dealing with head and other body parts separately, which we denote by $G^h$ and $G^b$, respectively. Both are an encoder-decoder network with skip connections but use different parametric values. As our training data is generated from a video that usually contains frames of different views and poses of the target person, the encoders can extract features of different poses. Furthermore, the appearance of the target is encoded in the network under the supervision of real images. Another generator, denoted by $G^f$, is designed to fuse the body parts generated by the above two generators and resized by the mid-output module into the final result $\hat{I}$ that it has the same pose of the input while preserves the appearance of the target. In the inference stage, the input of the network is a pose map extracted from the source image (video). The background used for composing the final output also comes from the source.

**Discriminators.** Corresponding to three generators, BPA-GAN employs three discriminators, denoted by $D^h$, $D^b$, and $D^f$, to justify their output $\hat{I}^h$, $\hat{I}^b$ and $\hat{I}$ separately. We directly adopt the discriminator described in PatchGAN [19] in our network.

**Loss functions.** Three loss functions $L_h$, $L_b$ and $L_f$ are respectively designed to the deviation of $\hat{I}^h$, $\hat{I}^b$ and $\hat{I}^h$ from their ground truth, respectively. And each of them is made up of three terms: data loss $L^x_{dat}$, perceptual loss $L^x_{per}$ and adversarial loss $L^x_{adv}$:

$$L^x = L^h_{l1} + \lambda_{per} L^h_{per} + \lambda_{adv} L^h_{adv}, x \in \{'h',' b',' f'\} \quad (4)$$

where $\lambda_{per}$ and $\lambda_{adv}$ are two blending coefficients.

The data loss measures the difference between the pixels of the predicted images and those of the ground-truth images. The perceptual loss penalizes the deviation of the predicted images and the ground-truth images in the VGG[32] feature space. The adversarial loss encourages the distribution of the predicted images to be close to the ground-truth.

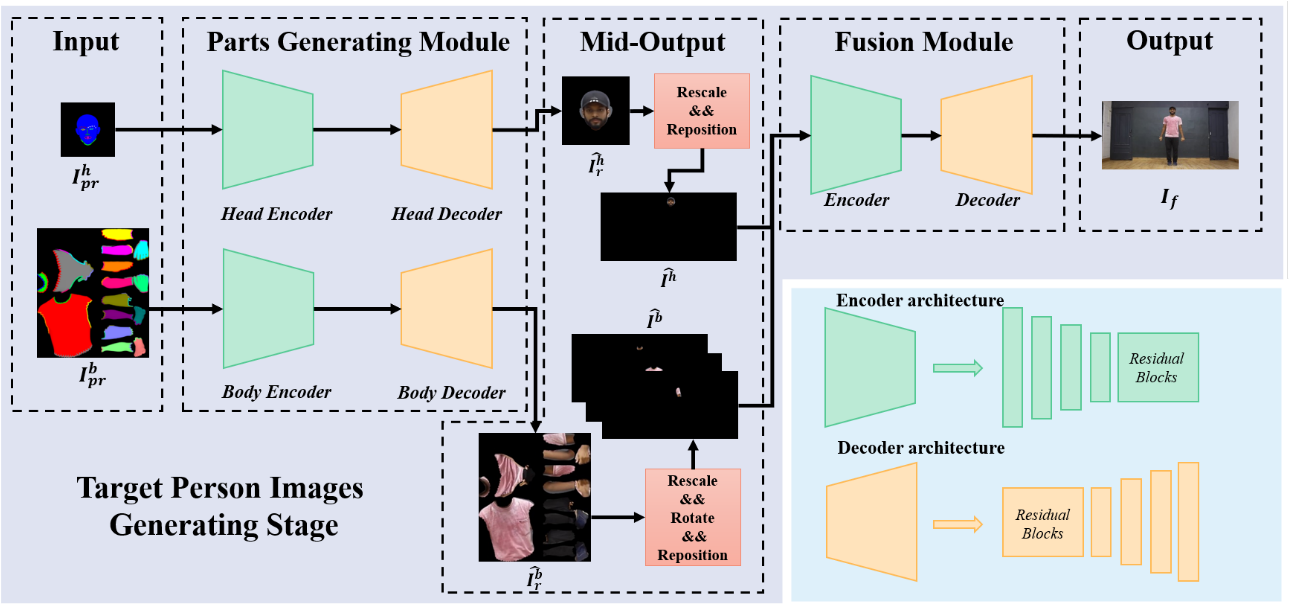


Figure 3. The architecture of BPA-GAN. The part generation module consists of two encoder-decoder structures which respectively take head pose map $I^h_{pr}$ and body pose map $I^b_{pr}$ as input,and correspondingly generate head image $\hat{I}^h_r$ and body image $\hat{I}^b_r$. The mid-output module recovers the body parts to their original size. And the fusion module combines the $\hat{I}^h$ and $\hat{I}^b$ to yield the final result $I_f$. Three discriminators are used to justify the generated head, body, and fused images, respectively.

They are respectively calculated as

$$L^x_{dat} = \left\| I^x_r - \hat{I}^x_r \right\|_1, \tag{5}$$

$$L^x_{per} = \left\| f_{VGG}(I^x_r) - f_{VGG}(\hat{I}_r) \right\|_1, \tag{6}$$

$$L^x_{adv} = \sum (D^x \left( I^x_r, \hat{I}^x_r \right) - 1)^2, x \in \{'h',' b',' f'\} \tag{7}$$

where $f_{VGG}$ denotes the VGG feature and $D^x$ represents one of the three discriminator $D^h$, $D^b$ and $D^f$. The formulas of $L_b$ and $L_f$ are similar with $L_h$. $L_b$ are the losses between the fake scaled body images $I_{f\_b\_r}$ and the target scaled body images $I_{t\_b\_r}$.

The discriminator losses are $L^h_D$, $L^b_D$, and $L^f_D$, which are designed for discriminators $D^h$, $D^b$, and $D^f$ respectively. These losses have the same form as that of LSGANs[26]. Nevertheless, we use the form in the PatchGAN[19] manner, namely, first uniformly partition the image into a set of small patches and then evaluate the loss patch by patch:

$$L^x_D = \sum D^x(I^x_{pr}, \hat{I}^h_r)^2 + \sum (D^x \left( I^x_{pr}, I^x_r \right) - 1)^2 \tag{8}$$

## 4. Results

**Datasets.** We downloaded 10 clips of dancing videos from the Internet among which the subjects in different videos are different. Furthermore, each subject wears different clothes. We call the subject in each video the target person. We randomly extracted 500 frames for each video and partition the 500 frames into the training set and the testing set with a ratio of $4:1$ without overlapping. Namely, we totally have 10 datasets.

We also created a cross-person test dataset containing 110 images among which 100 images are extracted from the above testing set with 10 images for per target person, and rest 10 images from the internet. This test dataset is used to quantitatively evaluate the transfer results between different people by our method. For more details, please see Subsection 4.1 and refer to the supplementary videos.

**Implementation details.** We first preprocessing the data as described in section 3.1. The encoders and decoders' architecture is the same as UNet [31], and we add three residual blocks behind the encoders and in the front of the decoders. The architecture of three discriminators is the same as PatchGAN [19]. While training, we set the batch size as 4, $\lambda_p$ as 10, $\lambda_r$ as 100. We use Adam [22] to optimize parameters of the generators and the discriminators with the hyperparameters $\beta_1 = 0.5$, $\beta_2 = 0.999$ and learning rate as 0.0002. To stabilize the training process, we train our end-to-end network in two stages. At the first stage, we only train the networks in the parts generating module till they converge. At the second stage, we fix the network parameters of the parts generating module and focus on training the fusion module. Empirically, such two-stages training leads to better performance.

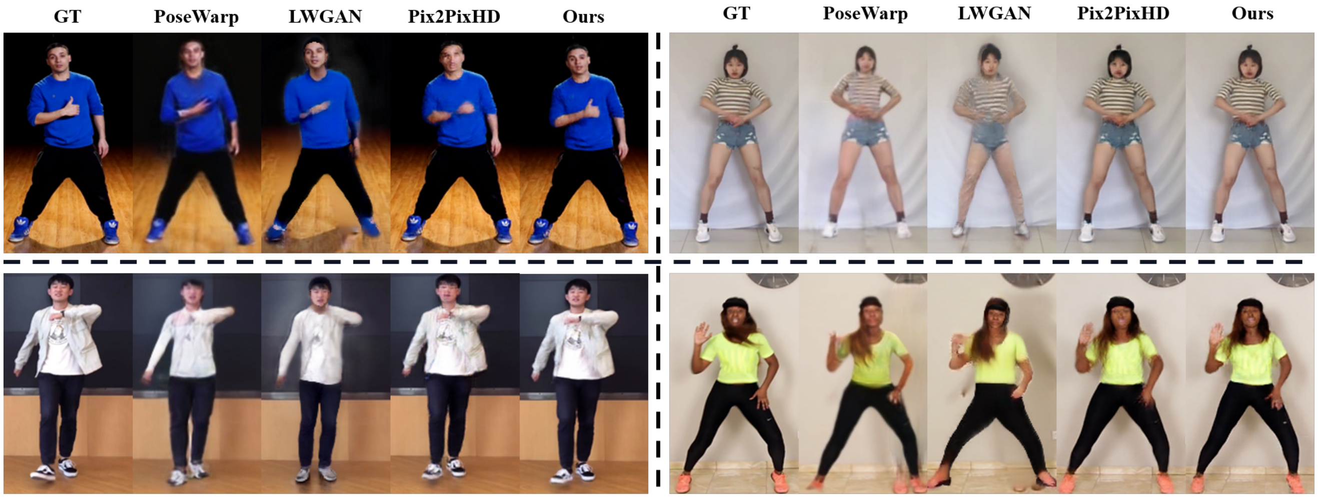


Figure 4. The qualitative results on the test datasets of our method and the other three methods. Our method performs better in detail preservation, such as hands and face. GT means the ground truth.

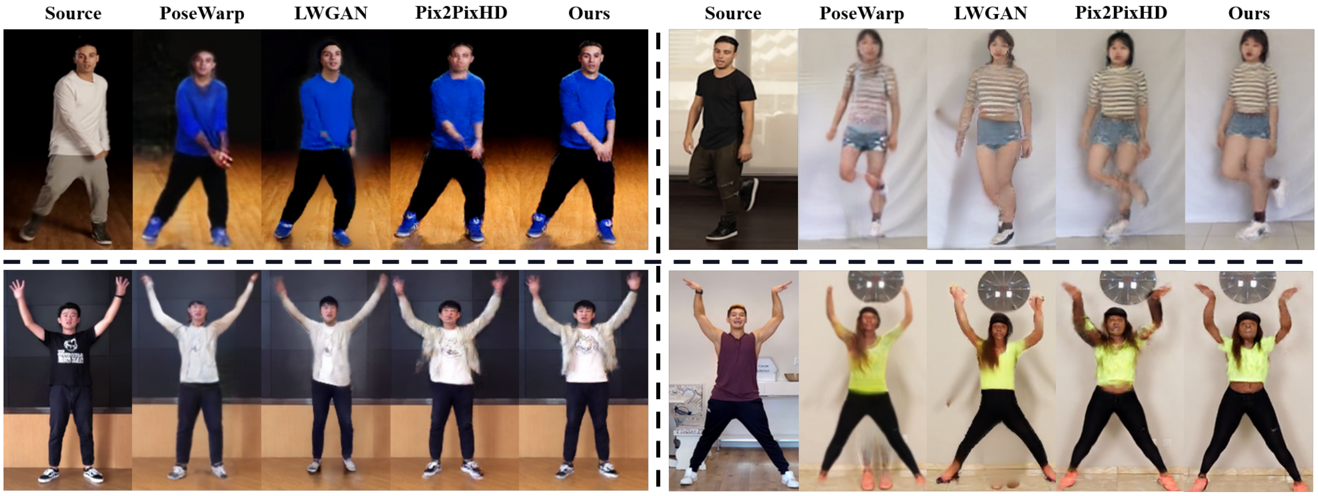


Figure 5. Comparison of motion transfer results. Our method outperforms than the other methods in the accuracy of the poses and the detail of the target person.

### 4.1. Quantitative results

We first evaluate the quantitative results which are obtained by using the test data which is extracted from the same video as the training dataset. This indicates that the source subject and target subject are the same. The evaluation metrics we used are SSIM, PSNR, LPIPS, and SSIM [39] among which PSNR measures the similarity between two images at the pixel level while LPIPS [44] measures the similarity between two images by the trained model.

We also quantitatively evaluate the transfer results from a source person different from the target one over the cross-person test dataset with 110 frames. We finally obtain 100 transfer results for each training dataset (correspondingly each target person) on the cross-person test dataset because we abandon 10 frames which are extracted from the same video as the training dataset. FID described in [16] is introduced to measure the similarity between two sets of images by the trained model. In our setting, we employ FID to measure the similarity between the set of cross-person transfer results and the ground-truth test dataset.

Our method is compared with PoseWarp [4], LWGAN [24] and pix2pixHD [38]. For PSNR and SSIM, the higher the score, the better the method. For LPIPS and FID, the lower the score, the better the method. As shown in Table 1, our method obviously outperforms existing approaches in all indices. Visually, our method also exhibits stronger ability to preserve the target person's features even in small parts like faces. This is because our GAN generates

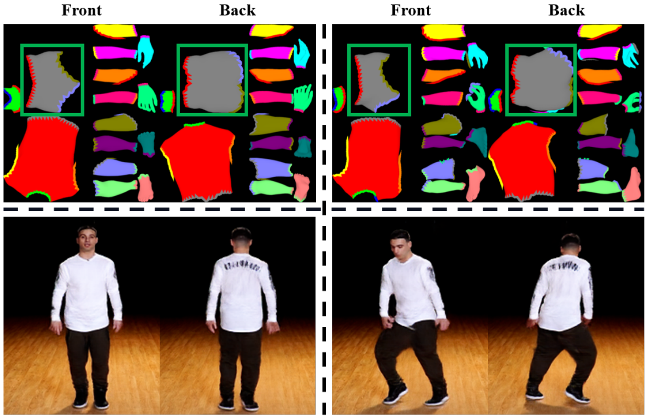


Figure 6. Images from the front side and back side of the same pose. The first row shows the pose images and the green boxes depict the body parts with significant differences. The second row shows the generated results.

the results part by part and therefore can sufficiently exploit the resolution of the original videos.

Table 1. Quantitative results of different methods and our method with different discriminators or different datasets. The higher score of PSNR and SSIM are, the better. The lower score of LPIPS and FID are, the better.

| | PSNR | SSIM | LPIPS | FID |
|---|---|---|---|---|
| PoseWarp[2018] | 29.368 | 0.795 | 0.115 | 134.547 |
| Pix2pixHD[2018] | 32.077 | 0.853 | 0.065 | 101.581 |
| LWGAN[2019] | 30.875 | 0.804 | 0.095 | 116.196 |
| Baseline | 35.176 | 0.888 | 0.045 | 79.377 |
| Baseline+H | 35.221 | 0.889 | 0.044 | 68.492 |
| Baseline+HB | 35.613 | 0.892 | 0.042 | 66.511 |
| **ours:Baseline+HBO** | **36.099** | **0.904** | **0.041** | **60.468** |

### 4.2. Qualitative results

The teaser figure shows some results of the motion transfer from the source person to the target person. We also make a qualitative evaluation of our method and the three methods mentioned above. In Figure 4, we show some results on our test datasets. We can see that our method outperforms the other methods with better details such as face and hands. In Figure 5, we transfer the pose of the source person into the target person. For the four cases, we have the best details on faces and bodies. The first row shows that we can better recover the details while the hands or legs are overlapped. The second row shows that our method has more accurate poses than other methods. We assign a specific color for each body part without explicitly marking the front and back of the body part. Fortunately, the extracted front and back body parts have slightly different shape. In addition, We will extract feature points for the front head while nothing is extracted for the back head. These make our approach have good ability to discriminate the front and back sides. Figure 6 shows that our approach can correctly generate transfer results for each side.

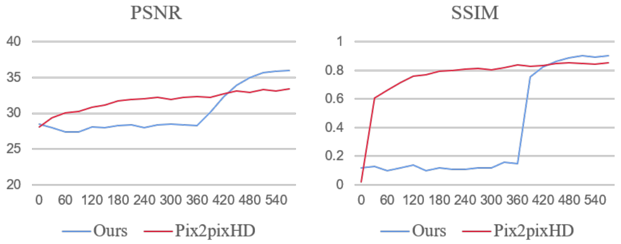


Figure 7. A quantitative comparison over time. The X-axis denotes training time in minutes. At the first stage, i.e., before 360 minutes, we train the part generating module only; therefore, the reconstruction error is high. Then, we start training the fusion module, where our method quickly outperforms Pix2pixHD in both PSNR and SSIM metrics.

### 4.3. Training efficiency

To demonstrate the training efficiency of BPA-GAN, we train both our method and Pix2pixHD on the same dataset and compare their performances over time, as shown in Figure 7. While training, we calculate the quantitative scores on the test set every 30 minutes. Because we train our parts generating module and fusion module one by one, our method's reconstruction error is high before 360 minutes, i.e., only training the first module. Once we begin to train the fusion module, our quantitative results improve rapidly and outperform the Pix2pixHD method.

### 4.4. Ablation studies

**Additional supervision.** We individually train our network with a different number of the discriminators to verify the additional supervision's impact. For our Baseline network, we only use fusion discriminator. Next, we add the supervision of the head discriminator for our Baseline+H networks. Then, we use all three discriminators for our Baseline+HB networks. Note that we train the three networks with the dataset without additional operations while rescaling. The quantitative results are shown in Table 1. As the number of discriminators grows, our trained model performs better and better. We can see the baseline results and the Baseline+H in Figure 8; the generated images have better head details while using head discriminator. And our Baseline+HB networks perform best with three discriminators.

**Datasets rescaling.** In section 3.1, we introduce the additional operation while rescaling the images. To verify the effectiveness of these operations, we train our network by the dataset without the operations, which means we rescale $I_{t\_h}$, $I_{p\_h}$, $I_{t\_b}$ and $I_{p\_b}$ into $512 \times 256$ directly. The quantitative results are shown in Table 1. Baseline+HBO means we train the networks with the datasets rescaled with additional operations like cutting for the head, cutting, and splicing body parts. The quantitative results show our Basline+HBO performs the best. In Figure 9, with the additional operations: 1) the red boxes of the first row show them help improve the results when the heads are deflected; 2) the red boxes of the second row show the network can generate better results when the organs of the faces are not clear; 3) the green boxes of the first row show the better detail of the body organs like hands; 4) the green boxed of the second row shows the better detail of the clothes like pants and belts.

**Model sequence smooth.** As we can see in Figure 11, the upper body and the right leg of the real frames are static. But in the pose guidance frames before smooth, the body's orientation and the right leg pose are not static, and the posture of left hands changes significantly in the third frame. With the constraints between frames, the body orientation, and the left leg pose, the pose guidance images keep static between frames. Besides, we can see that the left-hand motion is smoother. In Figure 10, we show the changes of the shape parameters and the pose parameters in the sequence. The left chart shows that the shape parameters are almost fixed. The right chart shows that the difference of the pose parameters between adjacent frames is smaller after smooth, suggesting that the model motion sequence is smoother.

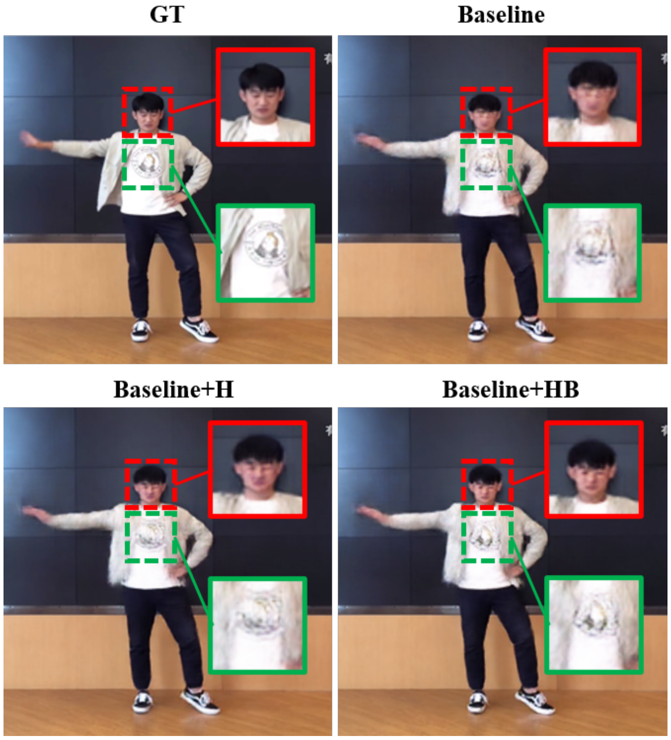


Figure 8. The results of the networks with different discriminator. GT denotes ground truth. Baseline denotes that we only use the fuse discriminator. Baseline+H means we add the head discriminator on the baseline. Baseline+HB means we use all three discriminators. The red boxes show the head details, and the green boxes show the body details.

## 5. Conclusion

This paper proposes BPA-GAN for human motion transfer. It introduces a body part map to represent human poses to support independently generating human body parts. Together with backgrounds, these parts are then seamlessly fused into an entire human body image with the specified pose via a fusing GAN. The part based mechanism has two merits. First, it can reduce mutual influence among different body parts during the composition of the new pose. Second, it enables partitioning a high-resolution image that contains the whole human body region into several moderate-resolution images as network input and therefore avoids compressing the high-resolution image to feed the network. To generate the body part map of an image, we use a 3D human model, which has been pre-segmented into 16 body parts, to fit the human pose in the image and then render the model using pseudo colors assigned to body parts. Extensive experiments show that our method outperforms the existing techniques to generate a more coherent human motion video and preserve more details of the target person.

The quality of the transfer results by the proposed method heavily depends on the accuracy of the body part map. Currently, we use a naked 3D human model as a geometric agent to segment the human body region in an image. For scenes in which a person wears loose clothes such as skirts, the quality the composite results will drastically decrease. It is also a challenge to deal with human motions with very large-scale pose changes, which we consider to tackle in future work. In addition, as having exerted smoothness constraints on adjacent poses when reconstructing the 3D mesh sequence, we get a smooth pose sequence as inputs of the network in the inference stage. Unfortunately, the target images used in the training stage are usually not adjacent. This makes it difficult to add constraints on the appearance of the network output. Therefore, the texture of our video results is not guaranteed temporally smooth. We would like to use the pre-trained models of some existing methods to improve our video results, as done in [7], and investigate how to improve the texture smoothness of the video even using a frame-by-frame training strategy.

## References

[1] K. Aberman, M. Shi, J. Liao, D. Lischinski, B. Chen, and D. Cohen-Or. Deep video-based performance cloning. In *Computer Graphics Forum*, volume 38, pages 219–233. Wiley Online Library, 2019. 1, 2, 3

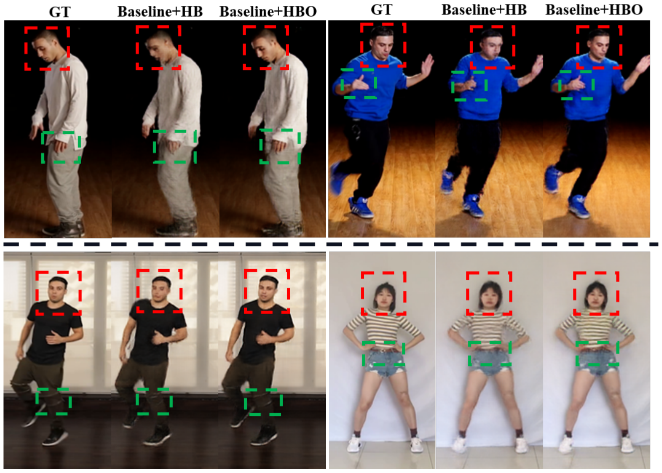


Figure 9. The results of the networks that are trained with different rescaled datasets. Baseline+HB means the networks trained with the datasets which are rescaled directly. Baseline+HBO means the networks trained with the datasets rescaled with additional operations like cutting or splicing. The red boxes show the improvement of the head, and the green boxes show the improvement in the body parts.

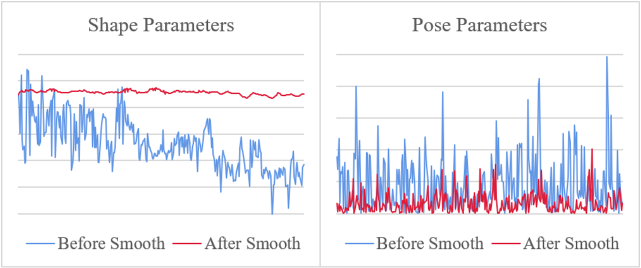


Figure 10. Comparison of the human model parameters before and after smoothing. The y-axis of the right chart means the sum of the absolute value of shape parameters, and the x-axis means the number of the frames in the sequence. The x-axis of the left chart means the difference of the sum of absolute values of pose parameters between adjacent frames, and the meaning of the x-axis is the same as the right chart.

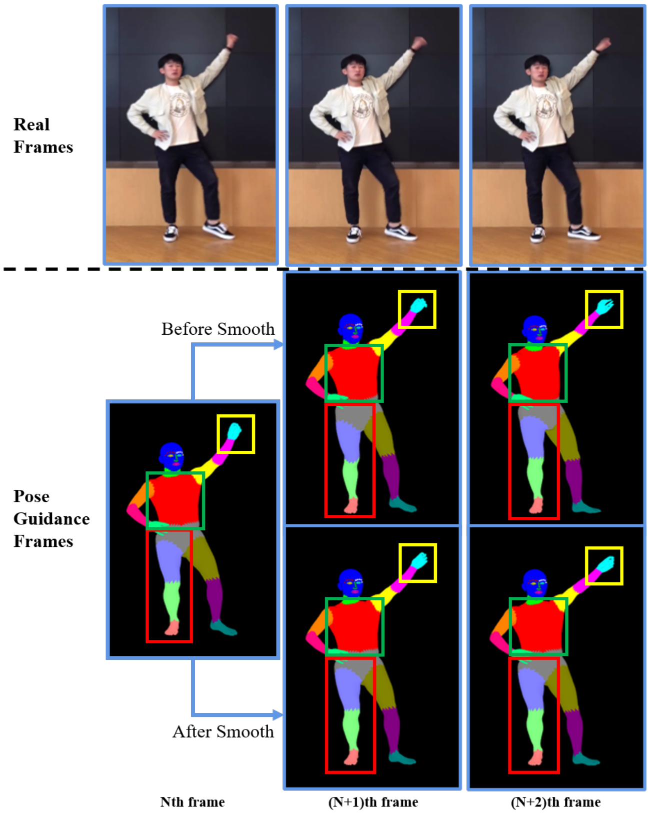


Figure 11. The model sequence frames comparison. The yellow box shows the pose of the left hand. The green box indicates the orientation of the body. The red box shows the pose of the right leg. The constraints between frames help the model sequence being temporally coherent.

[2] R. Alp Güler, N. Neverova, and I. Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7297–7306, 2018. 2

[3] A. Arnab, S. Zheng, S. Jayasumana, B. Romera-Paredes, M. Larsson, A. Kirillov, B. Savchynskyy, C. Rother, F. Kahl, and P. H. Torr. Conditional random fields meet deep neural networks for semantic segmentation: Combining probabilistic graphical models with deep learning for structured prediction. *IEEE Signal Processing Magazine*, 35(1):37–52, 2018. 2

[4] G. Balakrishnan, A. Zhao, A. V. Dalca, F. Durand, and J. Guttag. Synthesizing images of humans in unseen poses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8340–8348, 2018. 2, 7

[5] M. Barnard, M. Matilainen, and J. Heikkila. Body part segmentation of noisy human silhouette images. In *2008 IEEE International Conference on multimedia and expo*, pages 1189–1192. IEEE, 2008. 2

[6] R. Bellini, Y. Kleiman, and D. Cohen-Or. Dance to the beat: Synchronizing motion to audio. *Computational Visual Media*, 4(3):197–208, 2018. 2

[7] A. Chadha, J. Britto, and M. M. Roja. iseebetter: Spatio-temporal video super-resolution using recurrent generative back-projection networks. *Computational Visual Media*, 6(3):307–317, 2020. 9

[8] C. Chan, S. Ginosar, T. Zhou, and A. A. Efros. Everybody dance now. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5933–5942, 2019. 1, 2, 3

[9] S.-Y. Chen, W. Su, L. Gao, S. Xia, and H. Fu. Deepfacedrawing: Deep generation of face images from sketches. *ACM Transactions on Graphics (TOG)*, 39(4):72–1, 2020. 2

[10] K. Cheng, H.-Z. Huang, C. Yuan, L. Zhou, and W. Liu. Multi-frame content integration with a spatio-temporal attention mechanism for person video motion transfer. *arXiv preprint arXiv:1908.04013*, 2019. 2

[11] P. Esser, E. Sutter, and B. Ommer. A variational u-net for conditional appearance and shape generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8857–8866, 2018. 2

[12] K. Gong, X. Liang, Y. Li, Y. Chen, M. Yang, and L. Lin. Instance-level human parsing via part grouping network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 770–785, 2018. 2

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 2

[14] S. Guan, S. Wen, D. Yang, B. Ni, W. Zhang, J. Tang, and X. Yang. Human action transfer based on 3d model reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8352–8359, 2019. 2

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[16] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv preprint arXiv:1706.08500*, 2017. 7

[17] J.-W. Hsieh, C.-H. Chuang, S.-Y. Chen, C.-C. Chen, and K.-C. Fan. Segmentation of human body parts using deformable triangulation. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 40(3):596–610, 2010. 2

[18] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 3

[19] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 1, 2, 3, 5, 6

[20] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015. 3

[21] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018. 2

[22] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[23] L. Liu, W. Xu, M. Zollhoefer, H. Kim, F. Bernard, M. Habermann, W. Wang, and C. Theobalt. Neural rendering and reenactment of human actor videos. *ACM Transactions on Graphics (TOG)*, 38(5):1–14, 2019. 2, 3

[24] W. Liu, Z. Piao, J. Min, W. Luo, L. Ma, and S. Gao. Liquid warping gan: A unified framework for human motion imitation, appearance transfer and novel view synthesis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5904–5913, 2019. 1, 2, 7

[25] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015. 2

[26] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. On the effectiveness of least squares generative adversarial networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(12):2947–2960, 2018. 6

[27] N. Neverova, R. Alp Guler, and I. Kokkinos. Dense pose transfer. In *Proceedings of the European conference on computer vision (ECCV)*, pages 123–138, 2018. 2

[28] M. Omran, C. Lassner, G. Pons-Moll, P. Gehler, and B. Schiele. Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *2018 international conference on 3D vision (3DV)*, pages 484–494. IEEE, 2018. 2

[29] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10975–10985, 2019. 2, 3

[30] A. Ranjan, D. T. Hoffmann, D. Tzionas, S. Tang, J. Romero, and M. J. Black. Learning multi-human optical flow. *International Journal of Computer Vision*, pages 1–18, 2020. 2

[31] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2, 6

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[33] Y.-T. Sun, Q.-C. Fu, Y.-R. Jiang, Z. Liu, Y.-K. Lai, H. Fu, and L. Gao. Human motion transfer with 3d constraints and detail enhancement. *arXiv preprint arXiv:2003.13510*, 2020. 2, 3

[34] M. Wang, X.-Q. Lyu, Y.-J. Li, and F.-L. Zhang. Vr content creation and exploration with deep learning: A survey. *Computational Visual Media*, 6(1):3–28, 2020. 3

[35] M. Wang, G.-Y. Yang, R. Li, R.-Z. Liang, S.-H. Zhang, P. M. Hall, and S.-M. Hu. Example-guided style-consistent image synthesis from semantic labeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1495–1504, 2019. 2

[36] T.-C. Wang, M.-Y. Liu, A. Tao, G. Liu, J. Kautz, and B. Catanzaro. Few-shot video-to-video synthesis. *arXiv preprint arXiv:1910.12713*, 2019. 2

[37] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, G. Liu, A. Tao, J. Kautz, and B. Catanzaro. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*, 2018. 3

[38] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018. 2, 7

[39] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to

structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 7

[40] D. Wei, X. Xu, H. Shen, and K. Huang. Gac-gan: A general method for appearance-controllable human video motion transfer. *IEEE Transactions on Multimedia*, 2020. 2

[41] D. Xiang, H. Joo, and Y. Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10965–10974, 2019. 3

[42] W. Xu, A. Chatterjee, M. Zollhöfer, H. Rhodin, D. Mehta, H.-P. Seidel, and C. Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Transactions on Graphics (ToG)*, 37(2):1–15, 2018. 3

[43] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena. Self-attention generative adversarial networks. In *International Conference on Machine Learning*, pages 7354–7363, 2019. 2

[44] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 7

[45] L. Zhao, X. Peng, Y. Tian, M. Kapadia, and D. Metaxas. Learning to forecast and refine residual motion for image-to-video generation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 387–403, 2018. 2

[46] Y. Zhou, Z. Wang, C. Fang, T. Bui, and T. Berg. Dance dance generation: Motion transfer for internet videos. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3