Collaborative Neural Rendering Using Anime Character Sheets

Zuzeng Lin[1,2,*] , **Ailin Huang**[2,3,*] , **Zhewei Huang**[2,*]

[1]Tianjin University   [2]Megvii Technology   [3]Wuhan University

transpchan@gmail.com, {huangailin, huangzhewei}@megvii.com

https://github.com/megvii-research/IJCAI2023-CoNR

Abstract

Drawing images of characters with desired poses is an essential but laborious task in anime production. Assisting artists to create is a research hotspot in recent years. In this paper, we present the Collaborative Neural Rendering (CoNR) method, which creates new images for specified poses from a few reference images (AKA Character Sheets). In general, the diverse hairstyles and garments of anime characters defies the employment of universal body models like SMPL [Loper *et al.*, 2015], which fits in most nude human shapes. To overcome this, CoNR uses a compact and easy-to-obtain landmark encoding to avoid creating a unified UV mapping in the pipeline. In addition, the performance of CoNR can be significantly improved when referring to multiple reference images, thanks to feature space cross-view warping in a carefully designed neural network. Moreover, we have collected a character sheet dataset containing over $700,000$ hand-drawn and synthesized images of diverse poses to facilitate research in this area.

1 Introduction

2D Animation is one of the essential carriers of art reflecting human creativity. Artists commonly use character sheets to show their virtual character design. A character sheet is the image collection of a specific character with multiple postures observed from different views, as shown in Figure 1. It covers all the appearance details and is widely used to assist the creation of animations or their derived media. Moreover, character sheets allow many artists to cooperate while maintaining the consistency of the design of this character.

Drawing a sequence of anime frames is extremely time-consuming, requiring imagination and expertise. Due to the semantic gap between the character sheet and the desired poses, it is challenging to design a pipeline to draw character images automatically. Some non-photorealistic rendering (NPR) methods [Gooch and Gooch, 2001] can simulate the artistic style of hand-drawn animation. For example, Toon Shading is widely used in games and animation production.

However, it currently still requires a very complex manual design to approximate a specific artistic style. Artists need to manually retouch the shadows, which is too complicated for animators and painters. Therefore, we try to explore a new method of generating pictures in 2D animation.

We formulate the task of rendering a particular character in the desired pose from the character sheet. Based on this formulation, we develop a **Collaborative Neural Rendering (CoNR)** model based on convolutional neural network (CNN). CoNR fully exploits the information available in a provided set of reference images by using feature space cross-view dense correspondence. In addition, CoNR uses the Ultra-Dense Pose (UDP), an easy-to-construct compact landmark encoding tailored for anime characters. CoNR will not require a unified UV texture mapping [Yoon *et al.*, 2021; Gao *et al.*, 2020] in the pipeline, which may not be done in a consistent method for anime characters. UDP can represent the fine details of characters, such as accessories, hairstyles, or clothing, so as to allow better artistic control. It can also be easily generated with existing 3D computer graphics pipelines to adapt interactive applications. Moreover, we collect a character sheet dataset containing over $700,000$ hand-drawn and 3D-synthesized images of diverse poses and appearances. Training on this dataset, CoNR achieves impressive results both on hand-drawn and synthesized images. CoNR can help generate the character in the given pose. Creation of CoNR faithfully based on the character sheets, and it can quickly provide reference of the target pose for the artist. We also introduce how CNN-based UDP Detector generates UDPs from hand-drawn images in the Appendix. We provide a demo video in the supplementary material.

To sum up, our main contributions are:

- We formulate a new task, rendering 2D anime character images with desired poses using character sheets.

- We introduce a UDP representation for anime characters and collect a large character sheet dataset containing diverse poses and appearances. This dataset is made open-sourced.

- We explore a multi-view collaborative inference model, CoNR, to assist in producing impressive anime videos given action sequences specified by UDPs.

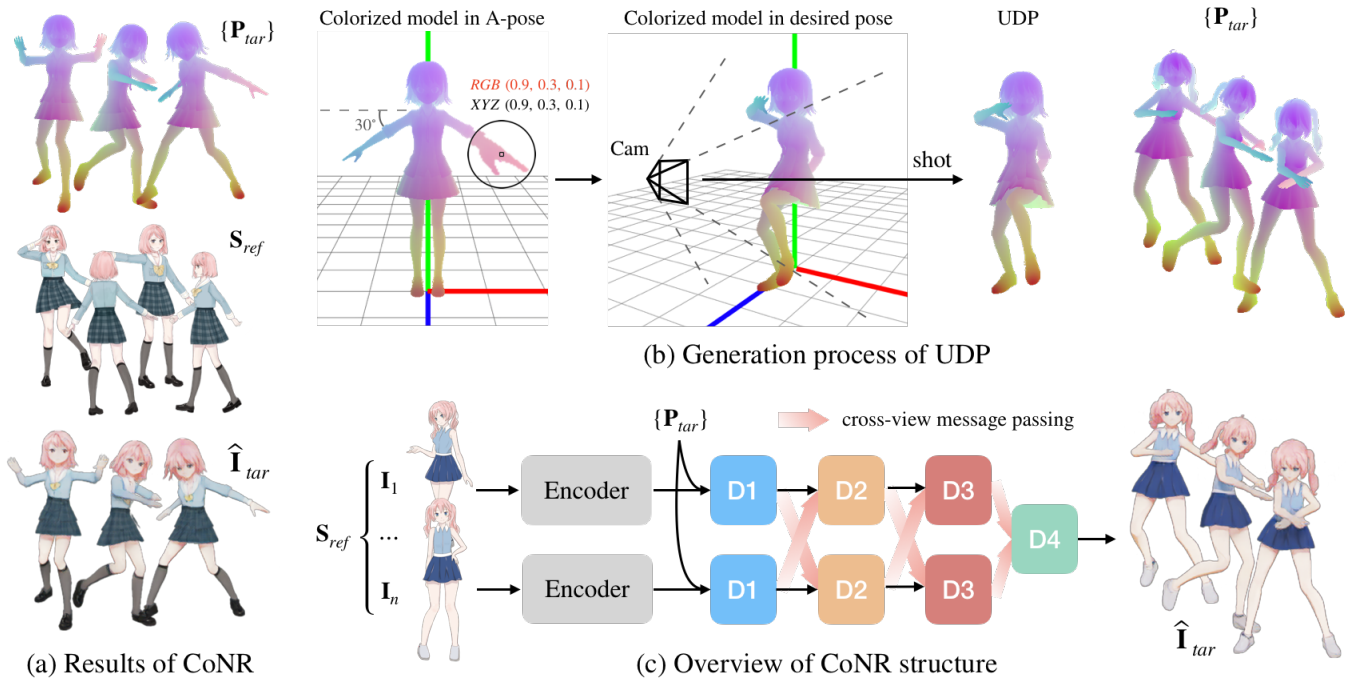*These three authors contribute equally.

Figure 1: **(a) The results of CoNR.** Based on the desired poses $\{\mathbf{P}_{tar}\}$ and the character sheet S_{ref}, CoNR renders new anime images $\hat{\mathbf{I}}_{tar}$. **(b) The generation process of UDP.** We use the XYZ coordinates of a point on the surface of a 3D model as the RGB value of the point and then color a 3D model. Then, we take a 2D view of the 3D model as UDP. **(c) Inference pipeline of CoNR.** Reference images $I_1 \cdots I_n \in \mathbf{S}_{ref}$ from the input character sheet are fed into CoNR using modified U-Nets [Ronneberger *et al.*, 2015] as sub-networks. UDP $\mathbf{P}_{tar}$ is resized and concatenated into each scale of the encoder outputs in all sub-networks. Blocks with the same color share weights. "D1 to D4" refers to four blocks of the decoder. Each block will receive the averaged message from corresponding blocks in all other sub-networks.

2 Related Work

2.1 Image Generation and Translation for Anime

Recent years have seen significant advancement in applying deep learning methods to assist the creation of anime. For example, Gao *et al.*; Zheng *et al.* propose to apply realistic lighting effects and shadow for 2D anime automatically; Wang and Yu propose to transfer photos to anime style; Siyao *et al.*; Chen and Zwicker propose frame interpolation tailored for animation. There are also attempts to produce vectorized anime images [Su *et al.*, 2021], similar to the step-by-step manual drawing process. The generative modeling of anime faces has achieved very impressive results [Jin *et al.*, 2017; Gokaslan *et al.*, 2018; Tseng *et al.*, 2020; Li *et al.*, 2021; He *et al.*, 2021]. The latent semantics of generative models has also been extensively explored [Shen and Zhou, 2021]. A modified StyleGAN2 model [aydao, 2021] can generate full-body anime images, although it still suffers from artifacts because of the high degree of freedom of the human body.

2.2 Representation of Human Body

Stick-figure of skeletons [Chan *et al.*, 2019], SMPL vectors [Loper *et al.*, 2015], and heat maps of joints [Cao *et al.*, 2019; Siarohin *et al.*, 2021] are widely-used representations obtained from motion capture system. However, when these sparse representations are used for anime characters [Khungurn and Chou, 2016], they face a new series of challenges: noisy manual annotations, unexpected occlusions caused by

the wide variety of characters' clothing and body shapes, and ambiguity due to hand drawing. Furthermore, the aforementioned human pose representations only represent human joints and body shapes. However, anime characters often require flexible artistic control over other body parts, such as the fluttering of hair and skirts. These representations cannot directly drive these parts.

Human parsers or clothing segmenters [Yoon *et al.*, 2021; Gafni *et al.*, 2021; Chou *et al.*, 2021] are robust to the uncertainty of joint positions. However, the provided semantic masks are not informative enough to represent the pose or even the orientation of a person. DensePose [Güler *et al.*, 2018] and UV texture mapping [Yao *et al.*, 2019; Yoon *et al.*, 2021; Gao *et al.*, 2020], greatly enhance the detail of pose representation on the human body or face by adding a universal definition that essentially unwarps the 3D human body surface into a 2D coordinate system. However, due to the great diversity and rich topology of anime characters, it is difficult to find a general labeling method to unwarp them, which makes existing dense representations still not serve as an off-the-shelf solution for anime-related tasks. One motivation for our work is to find a suitable representation to represent the motion of hair and clothes in principle.

There are also some dense representations of the body, such as CSE [Neverova *et al.*, 2020]. CSE addresses the task of detecting a continuous surface coding from an image. Similar "define-by-training" methods are inapplicable in anime creation since CSE can only be used for reenactment

currently.

2.3 Human Appearance Transfer

Most of works create vivid body motions or talking heads from only one single image [Gafni *et al.*, 2021; Sarkar *et al.*, 2020; Yoon *et al.*, 2021; Siarohin *et al.*, 2019; Liu *et al.*, 2019]. The learned prior of the human body [Loper *et al.*, 2015], head [Blanz and Vetter, 1999], or real-world clothing shape and textures [Alldieck *et al.*, 2019] enables the model to solve ill-posed problems like imagining and inpainting the back view even if only the frontal view is available. However, anime has long been featuring a flexible character design leading to high diversity in clothing, body shapes, hairstyles, and other appearance features. A model trained on a huge dataset (*e.g.*,, CLIP [Radford *et al.*, 2021]) might be able to encode some popular anime character designs implicitly. Still, it is generally more challenging to establish priors or styles of the anime character domain than the real human domain.

There are some attempts [Liu *et al.*, 2021] to extend the pose transfer task by utilizing SMPL [Loper *et al.*, 2015], a realistic 3D mesh of the naked human body that is based on skinning and blend-shapes, to combine appearance from different views. Using multiple reference images would, in principle, allow the model to follow the original appearance of the given person and better suit the needs of anime production.

Some recent works utilize neural rendering models [Mildenhall *et al.*, 2020] which are trained using photometric reconstruction loss and camera poses over multiple images of 3D objects and scenes. Due to their ray-marching nature and capability to in-paint in 3D, they are promising methods in modeling real-world 3D data [Peng *et al.*, 2021]. These methods are not influenced by or depend on prior knowledge other than the object to be modeled. However, they have not yet made much progress in modeling hand-drawn data like anime character sheets, which less follow strict geometric and physical constraints.

3 Method

3.1 Task Formulation

Our formulation is inspired by the tasks about drawing or painting art [Huang *et al.*, 2019; Su *et al.*, 2021]. We consider the character sheet $I_n \in S_{ref}$ in a whole. A target pose P_{tar} representation is also required to provide rendering target for the model. The task can be formulated as mapping S_{ref} to target image $\widehat{I}_{tar}$ with the desired target pose P_{tar}:

$$\widehat{I}_{tar} = \Phi(P_{tar}, S_{ref}). \qquad (1)$$

We notice that complicated poses, motions, or characters may require more references in S_{ref} than others, so dynamically sized S_{ref} should be allowed.

3.2 Ultra-Dense Pose

A UDP specifies a character's pose by mapping 2D viewport coordinates to feature vectors, which are 3-tuple floats that continuously and consistently encode body surfaces. In this way, a UDP can be represented as a color image $P_{tar} \in$

Figure 2: **Random characters with random backgrounds**.

$\mathbb{R}^{H \times W \times 3}$ with pixels corresponding to landmarks $L_{(x,y)} \in \mathbb{R}^3$. Non-person areas of the UDP image are masked. It allows better compatibility across a broader range of anime body shapes and enables better artistic control over body details like garment motions.

3D meshes are widely used data representations for anime characters in their game adaptations. Vertex in a mesh usually comprises corresponding texture coordinate (u, v) or a vertex color (r, g, b). Interpolation over the barycentric coordinates allows triangles to form faces filled with color values or pixels looked up from textures coordinates.

Taking a bunch of anime body meshes standing at the center of the world, we ask them to perform the same **A-pose** (a standard pose) to align the joints. To construct UDPs, we remove the original texture and overwrite the color (r, g, b) of each vertex with a landmark, which is currently the world coordinate (x, y, z), as shown in Figure 1(b). When the anime body changes its pose, the vertex on the mesh may move to a new position in the world coordinate system, but the landmark at the corresponding body part will remain the same color.

To avoid the difficulty of down-sampling and processing meshes, we convert the modified meshes into 2D images, which are compatible with CNNs. This is done by introducing a camera, culling on occluded faces, and projecting only the faces visible from the camera into an image. The processed UDP representation is a $H \times W \times 4$ shaped image recorded in floating-point numbers ranging from 0 to 1. The four channels include the three-dimensional landmark encodings and one-dimensional occupancy for indicating whether the pixel is on the body.

Three properties of UDP could alleviate the difficulties when creating images of 2D animation:

1) UDP is a detailed 3D pose representation since every piece of surface on the anime body could be automatically assigned with a unique encoding in 3D graphics software.

2) UDP is a compatible pose representation since the anime characters with similar body shapes will also get out-fits that are consistently pseudo-colorized.

3) UDP can be obtained directly in all existing 3D editors, game engines, and many other up-streams.

3.3 Data Preparation

As character sheets used in the anime-related industries are not yet available to the computer vision community, we built a dataset containing more than $20,000$ hand-drawn anime characters by selecting human-like characters from public

datasets [Anonymous *et al.*, 2021; Jerry, 2017]. We manually perform matting to remove the background from the character with the help of the watershed algorithm [Torralba *et al.*, 2010]. We also construct a 2D background dataset containing 4000 images with a similar method. Manually annotating hand-drawn anime images with UDP involves significant hardship. To alleviate the problem of label scarcity, we further construct a synthesized dataset from anime-styled 3D meshes.

Finally, We combine the synthesized dataset with the hand-drawn dataset to obtain both high-quality UDP labels and high diversity of hand-drawn styles. We randomly split the whole dataset by a $1/16$ ratio into the validation and training sets. The split is on a per-anime-character basis, so the validation set contains characters unseen during training. The whole dataset contains over $700,000$ hand-drawn and synthesized images of diverse poses and appearances. We manually exclude content containing excessive nudity that is not suitable for public display. Random characters with a random background are shown in Figure 2.

3.4 Collaborative Neural Rendering

Overview. We utilize a collaborative inference for a convolutional neural network named **CINN**, inspired by Point-Net [Qi *et al.*, 2017] and Equivariant-SFM [Moran *et al.*, 2021]. CoNR consists of a CINN renderer and an optional UDP Detector. Figure 1(c) shows the pipeline of the proposed approach. CoNR generates a character image of the desired pose, taking the UDP representation $\mathbf{P}_{tar}$ of the target pose and a character sheet $\mathbf{S}_{ref}$, as the inputs. When generating more than one pose, we feed different UDPs and use the same reference character sheet.

The input UDP representation can be produced by a UDP Detector from reference images. For interactive applications like games, the existing physics engine can be used as a drop-in replacement for the UDP Detector to compute body and cloth dynamics for the anime character directly.

Renderer. We apply the following modifications to the U-Net [Ronneberger *et al.*, 2015]:

1) As shown in Figure 1(c), we rescale and concatenate UDP to each block of the decoder. This input strategy aims to reuse the encoder's results and further allow highly efficient inference when there are multiple target UDPs when generating an animation video of a character.

2) Due to the spatial misalignment of the local and the remote branches, we use flow fields to align the features. Specifically, we approximate a flow field $\mathbf{f}$ as two channels and warp the features of other channels according to the estimated flow [Zhou *et al.*, 2016; Liu *et al.*, 2017; Hu *et al.*, 2023]. This operator enhances the long-range look-up ability for CINN.

3) We use the CINN method in the decoders of the network. We split the original up-sampling output feature channels by half, one for the remote branch and the other for the local branch. Firstly, we warp the features of the remote branches to align with the local features. Then the output features from all remote branches are averaged to be concatenated with the encoder output. The concatenated feature will be fed into the next block (illustrated in **Appendix**). Formally, to aggregate

local feature $Feat_l$ and remote feature $Feat_r^i (0 < i < k)$:

$$\mathbf{f}_r^i = Conv_{3\times3}(PReLU(Conv_{3\times3}(Feat_r^i))), \quad (2)$$

$$Feat_l^* = Feat_l + \sum_{i=0}^{k-1} \overleftarrow{\mathcal{W}}(Feat_r^i, \mathbf{f}_r^i)/k, \quad (3)$$

where we denote the pixel-wise backward warping (remapping) as $\overleftarrow{\mathcal{W}}$. $PReLU$ and $Conv_{3\times3}$ represent PReLU activation function [He *et al.*, 2015] and 3×3 convolution.

The last decoder block will collect averaged output features from all previous decoder blocks in all branches and output the final generated image $\widehat{I}_{tar}$.

UDP Detector. While training with the synthesized dataset, UDP $\widehat{\mathbf{P}}_{tar}$ can be directly obtained. As for the hand-drawn dataset, we design a UDP Detector to estimate UDPs $\widehat{\mathbf{P}}_{tar}$ from hand-drawn images. The UDP Detector is a simple U-Net [Ronneberger *et al.*, 2015] consists of a ResNet-50 (R50) [He *et al.*, 2016] encoder and a decoder with 5 residual blocks. It is trained jointly with the renderer in an end-to-end manner. We share the weight of the renderer's encoder and the UDP Detector's encoder.

4 Experiments

4.1 Training Strategy

We train CoNR with m sub-networks (views) on our dataset. To create one training sample, we randomly select a character and then randomly select $m+1$ arbitrary poses. These images are split as m image inputs (character sheet) $I_1 \cdots I_m \in S_{ref}$ and one image of the target pose as the ground truth of CoNR's final output.

We paste them onto k random backgrounds and feed them into the UDP Detector. We use the average of the k UDP detection results $\widehat{P}_i$ of the same target pose, $\widehat{P}_{tar} = 1/k \sum_{i=1}^{k} \widehat{P}_i$, as the final UDP detection results. We compute losses at both the output of the detector and the end of the CoNR pipeline. We use L1 loss on the landmark encodings and binary cross-entropy (BCE) loss on the mask to train the detector if the ground truth UDP P_{tar}^{GT} is available:

$$\mathcal{L}_{udp} = ||\widehat{P}_{tar} - P_{tar}^{GT}||_1, \quad (4)$$

$$\mathcal{L}_{mask} = BCE(sgn(\widehat{P}_{tar}), sgn(P_{tar}^{GT})), \quad (5)$$

where sgn indicates that a point is from the body surface or background. We use a consistency loss $\mathcal{L}_{cons}$ by computing the standard deviation of k UDP Detector outputs:

$$\mathcal{L}_{cons} = \sqrt{\frac{1}{k-1} \sum_{i=1}^{k} (\widehat{P}_i - \widehat{P}_{tar})^2}. \quad (6)$$

At the end of the collaborated renderer, we use L1 loss and feature loss [Ledig *et al.*, 2017] which is based on a pre-trained 19-layer VGG [Simonyan and Zisserman, 2015] network to supervise the reconstruction in the desired target pose, denoted as L_{photo} and L_{vgg}.

Setting	112K iter		224K iter	
	$\mathcal{L}_{photo}$	LPIPS	$\mathcal{L}_{photo}$	LPIPS
$m=1, n=1$	0.0247	0.0832	0.0238	0.0801
$m=4, n=1$	0.0249	0.0865	0.0237	0.0827
$m=1, n=4$	0.0219	0.0798	0.0211	0.0764
$m=4, n=4$	**0.0187**	**0.0659**	**0.0179**	**0.0612**

Table 1: **Comparison on the number of input reference images.** We use character sheets of m reference images to train the CoNR model, and then use character sheets of n reference images to evaluate the trained model.

Figure 3: **First row**: Inference results on validation dataset. **Second row**: Inference results with the same character sheet input S_{ref} on different body structure P_{tar}.

The UDP Detector and renderer are trained end-to-end simultaneously. The total loss function is the sum of all losses:

$$\mathcal{L} = \mathcal{L}_{udp} + \mathcal{L}_{cons} + \mathcal{L}_{mask} + \mathcal{L}_{photo} + \alpha\mathcal{L}_{vgg}, \quad (7)$$

where the hyper-parameter α of feature loss is from previous work [Zhang *et al.*, 2018b].

Our model is optimized by AdamW [Loshchilov and Hutter, 2019] with weight decay 10^{-4} for $224K$ iterations. We choose $m=4, k=4$ during training unless otherwise specified. The training process uses a batch size of 24 with all input resolutions set to 256×256. Model training takes about three weeks on four GPUs.

4.2 Result

For quantitative evaluation on the validation set, we measure L_{udp} and L_{photo} which is the averaged L1 distance between predictions and ground truth. We further use LPIPS [Zhang *et al.*, 2018a] to measure the perceptual quality.

Inference Visual Effects. CoNR can cope with the diverse styles of anime, including differences between synthesized and hand-drawn images. Figure 3 lists some random CoNR outputs on the validation split. With the same character sheet input S_{ref}, we replaced the provided UDP P_{tar}, and the results of CoNR changed accordingly. Notably, CoNR goes beyond a naïve correspondence by absolute UDP value (transfer between different cloths).
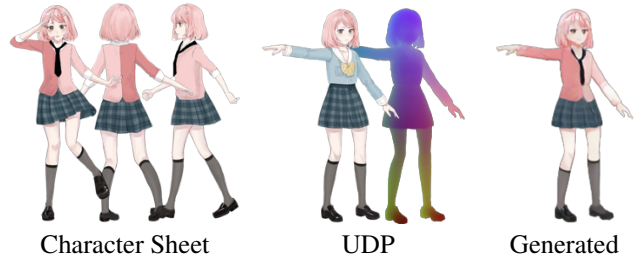

Character Sheet   UDP   Generated

Figure 4: **Transfer appearance based on the character sheet.** This UDP comes from UDP Detector rather than 3D softwares.

CoNR with UDP Detector can also be used to achieve similar results to style transfer methods when running a textured image at the reference pose. As shown in Figure 4, UDP detection of one character can be used to render another character. The CoNR inference pipeline for the anime (or game) production will usually be without the UDP Detector.

Effectiveness of the Collaboration. CoNR uses a dynamically-sized set of reference inputs during training and inference. Table 1 shows that using additional views ($m > 1$) during training will enhance the quality of generated images. On the opposite, removing images from the character sheet will reduce the coverage of the body surface. When CoNR is trained with $m = 1$, CoNR can not provide a reasonable solution. For example, given the character's backside, it is hard to imagine the frontal side. In this case, the photo-metric reconstruction and feature losses may encourage the network to learn a wrong solution. Therefore even if enough information in the $n > 1$ images is provided during inference, the network may not generate the target image accurately. Similarly, keeping $m = 4$ while reducing the number of inference views ($n = 1$) will also harm the quality of CoNR.

As shown in Figure 5, CoNR can leverage information distributed across all images to produce better visual effects. This allows CoNR to scale from image synthesis, when only a few shots of the character are given, to image interpolating when a lot of shots are available. The example in Figure 5 shows the behavior of CoNR when it does not have enough information to draw missing parts correctly. Even if very few reference images are provided, the target pose that is similar to some reference will be accurate. Furthermore, users may iterate on the results and feed them back into CoNR to accelerate anime production.

4.3 Comparison with Related Work

As CoNR provides a baseline for a new task, we admit that direct comparison with related works can either be impossible or unfair. We still try to include comparisons, only to show how our task is related to other existing tasks.

Human Pose Synthesis. We compare the results produced by CoNR to a real-world digital human system [Liu *et al.*, 2021] using the same target poses as used in their demo. We use two character sheets [1] with both pose and appearance unseen during training. One contains the same 4 images shown in

---

[1] One anime character is from www.youtube.com/watch?v=m6k_ t8yEyvE and another character is illustrated by an amateur artist.
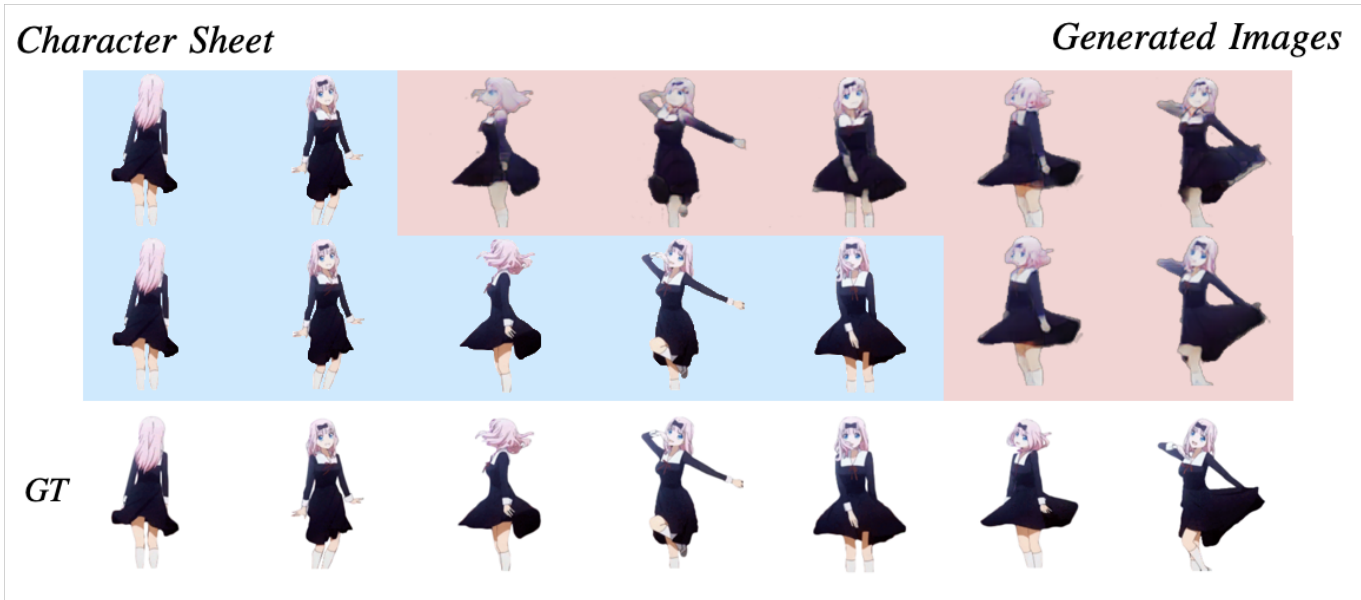
Figure 5: **Effectiveness of the collaboration.** We perform a reconstruction experiment with Chika Dance, which is a high-quality rotoscoping animation (in which body and clothing motions are drawn according to real characters) ensuring that the ground truth is reasonable. The last row shows 8 ground truth frames $I_i^{GT} \in S_{vid}^{GT}$ from a video. In this experiment, the UDPs are estimated from ground truth frames by a trained UDP Detector. The first two rows show the input and output images of CoNR. The used subsets of character sheet $S_{ref} \subset S_{vid}^{GT}$ are marked using the blue background. Generated images for novel poses are marked using the red backgrounds.

Setting	112K iter		224K iter	
	$\mathcal{L}_{udp}$	$\mathcal{L}_{mask}$	$\mathcal{L}_{udp}$	$\mathcal{L}_{mask}$
Original U-Net	0.1247	0.0856	Fail	Fail
U-Net+R34	0.1051	0.1068	0.1004	0.0747
U-Net+R50	**0.0969**	**0.0792**	**0.0971**	**0.0736**

Table 2: **Ablation on UDP Detector with different backbones**.

U-Net	Warping	CINN	R50	224K iter	
				$\mathcal{L}_{photo}$	LPIPS
✓				0.0311	0.1038
✓	✓			0.0308	0.1036
✓		✓	✓	0.0286	0.0977
✓	✓	✓		0.0180	0.0612
✓	✓	✓	✓	**0.0179**	**0.0612**

Table 3: **Ablation on Renderer.** The warping operation is performed among all branches.

Figure 1, the other character sheet taken from a random video from the internet as used in Figure 5. Figure 6 indicates that the long skirt prevents a high accuracy estimation of the leg joints and that parametric 3D human models like SMPL may not handle the body shape of anime characters correctly. Further diagnosis shows that parametric 3D human models may not handle anime's diverse clothing and body shape, as shown in Figure 2. CoNR can produce images at desired target poses with better quality.

Image synthesis pipelines starting with human pose representations are theoretically inapplicable on anime data, as the character's diverse body structure, clothing, and accessories cannot be reasonably represented. Using human pose synthesis methods in anime, we may lose artistic control over garments and fine details, which is crucial in anime creation workflows.

Style Transfer. Style transfer typically refers to applying a learned style from a certain domain (anime images) to the input image taken from the other domain (real-world images) [Chen *et al.*, 2019]. The models usually treat the target domain as a kind of style and require extensive training to remember the style. Some methods use a single image to

	Messaging	112K iter		224K iter	
		$\mathcal{L}_{photo}$	LPIPS	$\mathcal{L}_{photo}$	LPIPS
1 time	-	0.028	0.107	0.026	0.099
	1 time	0.019	0.066	0.018	0.063
3 times	3 times	**0.018**	**0.065**	**0.017**	**0.061**

Table 4: **Ablation on message passing**.

provide a style hint during the inference. For example, one could use Swapping Auto Encoders [Park *et al.*, 2020], a recent style-transfer method to swap the textures or body structures between two characters. Although the model has a lot of parameters (3× the size of CoNR), our comparison shows that it is still not enough to remember and reproduce the diverse sub-styles of the textures, pose, and body structure in the domain of anime.
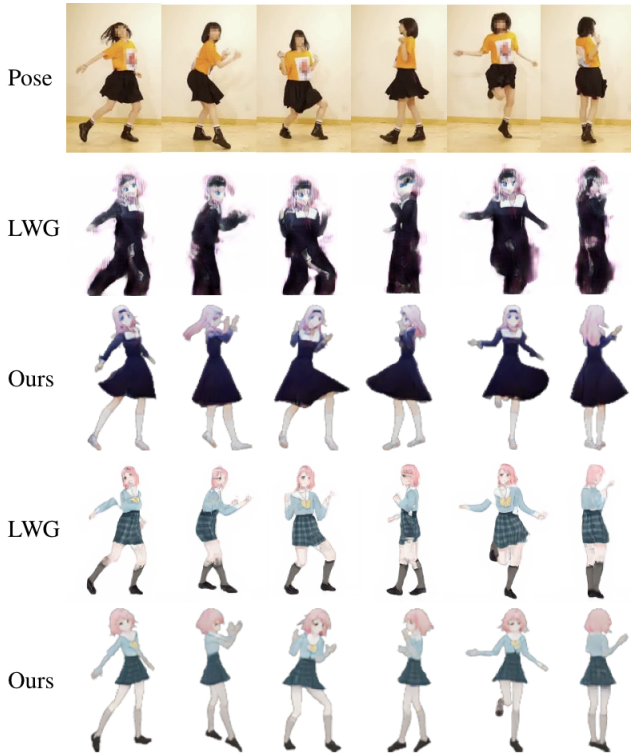
Figure 6: **Conceptual comparison with pure image-based method.** We compare the results of CoNR with the results of Liquid Warping GAN (LWG) [Liu *et al.*, 2021] when trying to resemble the target poses. We use 3D model editor software to generate pose-related UDPs based on the real person then make CoNR render the texture. And LWG uses the real person images as input.

4.4 Ablation Study

We perform ablation studies on the UDP Detector, renderer, loss functions and message passing. Table 2 shows that UDP representation can be inferred from images using a U-Net [Ronneberger *et al.*, 2015]. An original U-Net, which takes a concatenated tensor of 4 reference images and the target UDP as the input, does not provide acceptable results on this task, as shown in Table 3. The proposed CoNR method with both the feature warping and the CINN method significantly increases the performance, thus establishing a stronger baseline for the proposed task. We further ablate the loss functions in Table 5. In Table 4, we perform ablations on the number of message passing for the CoNR. In default, the subnetworks communicate at three different feature levels. More ghosting can be observed when sub-networks communicate less than three times.

5 Limitations

The inputs of CoNR can not provide any information about the environmental or contextual information that could be utilized to infer lighting effects. Users may have to look for sketch relighting techniques [Zheng *et al.*, 2020]. The generation results of finer layering and structuring are also to be studied.

Figure 7: **Comparision between detection results of hand-drawn anime character images using OpenPose [Cao *et al.*, 2019], the UDP Detector and SMPLify [Bogo *et al.*, 2016].** The images are from the validation split of the hand-drawn dataset [Anonymous *et al.*, 2021] with the ID number. The detected SMPL body can not fully handle the diverse body shapes of anime characters. UDP Detector produces relatively reasonable results.

Setting	224K iter			
	$\mathcal{L}_{udp}$	$\mathcal{L}_{mask}$	$\mathcal{L}_{photo}$	LPIPS
w/o $\mathcal{L}_{mask}$	0.162	0.880	-	-
w/o $\mathcal{L}_{photo}$	-	-	0.023	0.084
w/o $\mathcal{L}_{vgg}$	-	-	0.028	0.158
CoNR	**0.097**	**0.079**	**0.019**	**0.066**

Table 5: **Ablation on loss functions.**

CoNR is unable to model the dynamics of the character. The CoNR model accepts target poses detected from a video of a character with a body shape similar to S_{ref}, and could inherit the dynamics. However, it requires such a pose sequence beforehand. To bypass the UDP Detector, we can rely on additional technologies like garment captures [Bradley *et al.*, 2008], physics simulations [Baraff and Witkin, 1998], learning-based methods [Tiwari *et al.*, 2021] or existing 3D animation workflows to obtain a synthesized UDP P_{tar}. CoNR focuses on the rendering task. Obtaining a suitable 3D mesh with all the body parts rigged and all clothing computed with proper dynamics is beyond the scope of this paper. Using P_{tar} from an inappropriate body shape may lead to incorrect CoNR results.

The dataset may not fully follow the distribution of anime characters in the wild. The collected dataset contains only human-like anime characters from 2014 to 2018. As the character meshes are aligned using joints, the models trained on this dataset may not be applied to animal-like characters. Research on broader datasets will be the future work.

6 Conclusion

In this paper, we explore a new task to render anime character images with the desired pose from multiple images in character sheets. We develop a UV-mapping-free method, CoNR, achieving encouraging effectiveness. We show the potential of this data-driven method in assisting animation art. We hope the method and the datasets presented in this paper will inspire our community and further researchers.

References

[Alldieck *et al.*, 2019] Thiemo Alldieck, Gerard Pons-Moll, Christian Theobalt, and Marcus Magnor. Tex2shape: Detailed full human body geometry from a single image. In *International Conference on Computer Vision (ICCV)*, 2019. 3

[Anonymous *et al.*, 2021] Anonymous, Danbooru community, and Gwern Branwen. Danbooru2020: A large-scale crowdsourced and tagged anime illustration dataset. https://www.gwern.net/Danbooru2020, January 2021. Accessed: DATE. 4, 7, 10

[aydao, 2021] aydao. Website: This anime does not exist. https://thisanimedoesnotexist.ai, 2021. 2

[Baraff and Witkin, 1998] David Baraff and Andrew Witkin. Large steps in cloth simulation. In *25th annual conference on Computer graphics and interactive techniques*, pages 43–54, 1998. 7

[Blanz and Vetter, 1999] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999. 3

[Bogo *et al.*, 2016] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *European conference on computer vision (ECCV)*, 2016. 7

[Bradley *et al.*, 2008] Derek Bradley, Tiberiu Popa, Alla Sheffer, Wolfgang Heidrich, and Tamy Boubekeur. Markerless garment capture. In *ACM SIGGRAPH*, 2008. 7

[Cao *et al.*, 2019] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 43(1):172–186, 2019. 2, 7

[Chan *et al.*, 2019] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *International Conference on Computer Vision (ICCV)*, 2019. 2

[Chen and Zwicker, 2022] Shuhong Chen and Matthias Zwicker. Improving the perceptual quality of 2d animation interpolation. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[Chen *et al.*, 2019] Jie Chen, Gang Liu, and Xin Chen. Animegan: A novel lightweight gan for photo animation. In *International Symposium on Intelligence Computation and Applications*, pages 242–256. Springer, 2019. 6

[Chou *et al.*, 2021] Chien-Lung Chou, Chieh-Yun Chen, Chia-Wei Hsieh, Hong-Han Shuai, Jiaying Liu, and Wen-Huang Cheng. Template-free try-on image synthesis via semantic-guided optimization. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 2021. 2

[Gafni *et al.*, 2021] Oran Gafni, Oron Ashual, and Lior Wolf. Single-shot freestyle dance reenactment. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3

[Gao *et al.*, 2018] Zhengyan Gao, Taizan Yonetsuji, Tatsuya Takamura, Toru Matsuoka, and Jason Naradowsky. Automatic Illumination Effects for 2D Characters. In *NIPS Workshop on Machine. Learning for Creativity and Design.*, 2018. 2

[Gao *et al.*, 2020] Chen Gao, Yichang Shih, Wei-Sheng Lai, Chia-Kai Liang, and Jia-Bin Huang. Portrait neural radiance fields from a single image. *arXiv preprint arXiv:2012.05903*, 2020. 1, 2

[Gokaslan *et al.*, 2018] Aaron Gokaslan, Vivek Ramanujan, Daniel Ritchie, Kwang In Kim, and James Tompkin. Improving shape deformation in unsupervised image-to-image translation. In *European Conference on Computer Vision (ECCV)*, 2018. 2

[Gooch and Gooch, 2001] Bruce Gooch and Amy Gooch. *Non-photorealistic rendering*. AK Peters/CRC Press, 2001. 1

[Güler *et al.*, 2018] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[He *et al.*, 2015] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *International Conference on Computer Vision (ICCV)*, 2015. 4

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 4

[He *et al.*, 2021] Zhenliang He, Meina Kan, and Shiguang Shan. Eigengan: Layer-wise eigen-learning for gans. In *International Conference on Computer Vision (ICCV)*, 2021. 2

[Hu *et al.*, 2023] Xiaotao Hu, Zhewei Huang, Ailin Huang, Jun Xu, and Shuchang Zhou. A dynamic multi-scale voxel flow network for video prediction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 4

[Huang *et al.*, 2019] Zhewei Huang, Shuchang Zhou, and Wen Heng. Learning to Paint With Model-Based Deep Reinforcement Learning. In *International Conference on Computer Vision (ICCV)*, 2019. 3

[Jerry, 2017] Li Jerry. Pixiv dataset. https://github.com/jerryli27/pixiv_dataset, 2017. 4, 10

[Jin *et al.*, 2017] Yanghua Jin, Jiakai Zhang, Minjun Li, Yingtao Tian, Huachun Zhu, and Zhihao Fang. Towards the Automatic Anime Characters Creation with Generative Adversarial Networks. In *NIPS Workshop on Machine. Learning for Creativity and Design.*, 2017. 2

[Khungurn and Chou, 2016] Pramook Khungurn and Derek Chou. Pose estimation of anime/manga characters: a case for synthetic data. In *1st International Workshop on coMics ANalysis, Processing and Understanding*, pages 1–6, 2016. 2, 10

[Ledig *et al.*, 2017] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4

[Li *et al.*, 2021] Minjun Li, Yanghua Jin, and Huachun Zhu. Surrogate gradient field for latent space manipulation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[Liu *et al.*, 2017] Ziwei Liu, Raymond A Yeh, Xiaoou Tang, Yiming Liu, and Aseem Agarwala. Video frame synthesis using deep voxel flow. In *International Conference on Computer Vision (ICCV)*, 2017. 4

[Liu *et al.*, 2019] Wen Liu, Zhixin Piao, Jie Min, Wenhan Luo, Lin Ma, and Shenghua Gao. Liquid warping gan: A unified framework for human motion imitation, appearance transfer and novel view synthesis. In *International Conference on Computer Vision (ICCV)*, 2019. 3

[Liu *et al.*, 2021] Wen Liu, Zhixin Piao, Zhi Tu, Wenhan Luo, Lin Ma, and Shenghua Gao. Liquid warping gan with attention: A unified framework for human image synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2021. 3, 5, 7

[Loper *et al.*, 2015] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015. 1, 2, 3

[Loshchilov and Hutter, 2019] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. 5

[Mildenhall *et al.*, 2020] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision (ECCV)*, 2020. 3

[Moran *et al.*, 2021] Dror Moran, Hodaya Koslowsky, Yoni Kasten, Haggai Maron, Meirav Galun, and Ronen Basri. Deep permutation equivariant structure from motion. In *International Conference on Computer Vision (ICCV)*, 2021. 4

[Neverova *et al.*, 2020] Natalia Neverova, David Novotny, Marc Szafraniec, Vasil Khalidov, Patrick Labatut, and Andrea Vedaldi. Continuous surface embeddings. In *Advances in Neural Information Processing Systems (NIPS)*, 2020. 2

[Park *et al.*, 2020] Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei Efros, and Richard Zhang. Swapping autoencoder for deep image manipulation. In *Advances in Neural Information Processing Systems (NIPS)*, 2020. 6, 10, 11

[Peng *et al.*, 2021] Sida Peng, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Xiaowei Zhou, and Hujun Bao. Animatable neural radiance fields for modeling dynamic human bodies. In *International Conference on Computer Vision (ICCV)*, 2021. 3

[Qi *et al.*, 2017] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *IEEE Conference on Computer Vision and pattern recognition (CVPR)*, 2017. 4

[Radford *et al.*, 2021] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 3

[Raj *et al.*, 2021] Amit Raj, Julian Tanke, James Hays, Minh Vo, Carsten Stoll, and Christoph Lassner. Anr: Articulated neural rendering for virtual avatars. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 10

[Ronneberger *et al.*, 2015] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention (MICCAI)*, 2015. 2, 4, 7

[Sarkar *et al.*, 2020] Kripasindhu Sarkar, Dushyant Mehta, Weipeng Xu, Vladislav Golyanik, and Christian Theobalt. Neural re-rendering of humans from a single image. In *European Conference on Computer Vision (ECCV)*, 2020. 3

[Shen and Zhou, 2021] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[Siarohin *et al.*, 2019] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. In *Advances in Neural Information Processing Systems (NIPS)*, 2019. 3

[Siarohin *et al.*, 2021] Aliaksandr Siarohin, Oliver J Woodford, Jian Ren, Menglei Chai, and Sergey Tulyakov. Motion representations for articulated animation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[Simonyan and Zisserman, 2015] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015. 4

[Siyao *et al.*, 2021] Li Siyao, Shiyu Zhao, Weijiang Yu, Wenxiu Sun, Dimitris Metaxas, Chen Change Loy, and Ziwei Liu. Deep animation video interpolation in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[Su *et al.*, 2021] Hao Su, Jianwei Niu, Xuefeng Liu, Jiahe Cui, and Ji Wan. Vectorization of raster manga by deep reinforcement learning. *arXiv preprint arXiv:2110.04830*, 2021. 2, 3

[Tiwari *et al.*, 2021] Garvita Tiwari, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. Neural-gif: Neural generalized implicit functions for animating people in clothing. In *International Conference on Computer Vision (ICCV)*, 2021. 7

[Torralba *et al.*, 2010] Antonio Torralba, Bryan C Russell, and Jenny Yuen. Labelme: Online image annotation and applications. *Proceedings of the IEEE*, 98(8):1467–1484, 2010. 4

[Tseng *et al.*, 2020] Hung-Yu Tseng, Matthew Fisher, Jingwan Lu, Yijun Li, Vladimir Kim, and Ming-Hsuan Yang. Modeling artistic workflows for image generation and editing. In *European Conference on Computer Vision (ECCV)*, 2020. 2

[Wang and Yu, 2020] Xinrui Wang and Jinze Yu. Learning to cartoonize using white-box cartoon representations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[Yao *et al.*, 2019] Pengfei Yao, Zheng Fang, Fan Wu, Yao Feng, and Jiwei Li. Densebody: Directly regressing dense 3d human pose and shape from a single color image. *arXiv preprint arXiv:1903.10153*, 2019. 2

[Yoon *et al.*, 2021] Jae Shin Yoon, Lingjie Liu, Vladislav Golyanik, Kripasindhu Sarkar, Hyun Soo Park, and Christian Theobalt. Pose-guided human animation from a single image in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 2, 3

[Zhang *et al.*, 2018a] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 5

[Zhang *et al.*, 2018b] Xuaner Zhang, Ren Ng, and Qifeng Chen. Single image reflection separation with perceptual losses. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 5

[Zheng *et al.*, 2020] Qingyuan Zheng, Zhuoru Li, and Adam Bargteil. Learning to shadow hand-drawn sketches. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 7

[Zhou *et al.*, 2016] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. View synthesis by appearance flow. In *European Conference on Computer Vision (ECCV)*, 2016. 4

7 Appendix

7.1 Postscript

Zuzeng Lin explored the idea of assisting anime creation with neural networks. He proposed CoNR as a baseline to solve consistency and artistic control issues at the end of the 2020. He did most of the experiments and wrote the first draft. He quit subsequent submissions because of receiving the negative reviews from CVPR2022 and ECCV2022. He was therefore no longer involved in the submissions to AAAI2023 and IJCAI2023, which were made by other authors. They revised the paper to its present state with various demos. Zuzeng appreciates the discussions with many people who are interested in this project and Live3D public beta users in Sept. 2021.

7.2 Dataset Details

The synthesized part of the dataset contains multiple-pose and multiple-view images. Each of them is paired with UDP annotations. A randomly-sampled subset of 3D mesh data obtained in the same way as in [Khungurn and Chou, 2016] was used in this work, which contains $2,000$ anime characters with more than 200 poses. The scope of permissions was confirmed before using existing assets in this work. The dataset contribution in this work is mostly about years of manual data clean-up, which includes converting mesh formats, restructuring the bone hierarchy, and aligning the A-poses before synthesizing RGB images for each pose, and their corresponding UDPs with MikuMikuDance (MMD) software.

The final dataset also contains unlabeled hand-drawn samples and one-shot samples, *i.e.* characters with the only single available pose, which are randomly sampled from existing datasets [Anonymous *et al.*, 2021; Jerry, 2017]. We keep the licenses of all 3D models and motions meet the usage and distribution requirements.

7.3 Experiment Details

We make full use of the dataset by performing semi-supervised learning. For the UDP Detector, we skip the $\mathcal{L}_{udp}$ when we do not have ground truth of the UDP. We will keep the $\mathcal{L}_{cons}$ to encourage the UDP Detector to make a same prediction of one character under k random augmentations.

As for the renderer, we have to reuse the image of the same pose multiple times to fill all the m views if the provided images in the character sheet are not enough. Random-crop augmentation is applied to $\mathbf{I}_m \in \mathbf{S}_{ref}$, $\widehat{\mathbf{P}}_{tar}$ and $\mathbf{I}_{tar}^{gt}$. With the random-crop augmentation, in order to generate the image of body parts in the right position, CoNR has to learn a cross-modality feature matching instead of a trivial solution by simply selecting one of the provided images as the output.

7.4 Comparison with Style Transfer

To best of our knowledge, fusion of pose and texture for virtual character cannot be solved directly by implicit style transfer. As shown in Figure 8, SwapAE [Park *et al.*, 2020] fails in this scene.

7.5 Relationship with Neural Digital Human System

Some neural rendering systems can produce demos which may be similar to CoNR visually. For example, ANR [Raj *et al.*, 2021] uses UV mapping, which requires the assumption that the characters essentially have a similar topology. The main difference is that CoNR may deal with more general situations such as hair and long skirts. In addition, real-world video is constrained by geometric consistency, but painting creation is not. For 2D animation, the existing methods can not even accurately identify the skeleton. This domain gap is a part of our motivation to explore UDP tailored for animation.

Figure 8: **Evaluation results of Swapping Autoencoder (SwapAE) [Park *et al.*, 2020] for Deep Image Manipulation.** We trained a PyTorch implementation of SwapAE [Park *et al.*, 2020] with pairs of images of different characters in our dataset. This figure, from the first to the last row, shows (a) the target pose image, (b) the reference image, (c) pose of **a** and texture of **b**, fused by SwapAE, (d) pose of **b** and texture of **a**, fused by SwapAE.