This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2024.3375330

Hierarchical Feature Warping and Blending for Talking Head Animation

Jiale Zhang, Chengxin Liu, Ke Xian, and Zhiguo Cao

***Abstract*—Talking head animation transforms a source anime image to a target pose, where the transformation includes the change of facial expression and head movement. In contrast to existing approaches that operate on the low-resolution image (256×256), we study this task at a higher resolution, *e.g.*, 512×512. High-resolution talking head animation, however, raises two major challenges: i) how to achieve smooth global transformation while maintaining rich details of anime characters under large-displacement pose variations; ii) how to address the shortage of data, because no related dataset is publicly available. In this paper, we present a Hierarchical Feature Warping and Blending (HFWB) model, which tackles talking head animation hierarchically. Specifically, we use low-level features to control global transformation and high-level features to determine the details of anime characters, under the guidance of feature flow fields. These features are then blended by selective fusion units, outputting transformed anime images. In addition, we construct an anime pose dataset—AniTalk-2K, aiming to alleviate the shortage of data. It contains around 2000 anime characters with thousands of different face/head poses at a resolution of 512×512. Extensive experiments on AniTalk-2K demonstrate the superiority of our approach in generating high-quality anime talking heads over state-of-the-art methods.**

***Index Terms*—Talking head animation, generative adversarial networks, pose transformation, anime image generation, anime dataset**

I. Introduction

TALKING head animation, aiming at generating target anime talking heads with the change of facial expression and head pose for source anime characters, is an interesting but challenging problem with broad applications, such as virtual avatars, game production, and film making. Actually, there has been significant progress in the generation of human talking heads since the introduction of Generative Adversarial Networks (GANs) [1]. Given source human image, existing methods typically rely on facial landmarks [2], [3], [4], [5], [6] or audio [7], [8], [6] to generate human talking heads. Another related task is facial attribute editing [9], [10]. It manipulates source human image to possess desired attributes, such as facial expressions, hair color, and age. Despite recent progress on human talking head generation, less effort has been made on anime talking head generation. CPTNet [11] and Khungurn [12], for instance, tackle anime talking heads using a single image and a target pose vector. However, they only operate on the low resolution anime images (256×256), and can merely tackle small-displacement head motion with no occlusion. In addition, MakeItTalk [13] receives audio as input to generate anime talking heads, but it still limits to 256×256 resolution. We shall note that anime talking heads differ from human talking heads in several aspects: i) anime talking heads tend to have richer color and more regular textures than human; ii) the landmarks of human face can be easily acquired to guide the generation of human talking heads, while anime landmarks are hard to obtain; iii) there exist substantial human face datasets while high-quality anime pose datasets are lacking. Furthermore, when it comes to high resolution anime image, another challenge arises: *how to tackle large-displacement anime head movement with heavy occlusion while preserving the rich details of high-quality anime images?*

In this paper, given an input anime image and a target pose vector, our goal is to generate vivid high-quality talking heads for anime characters at a higher resolution of 512×512. To this end, we present a hierarchical feature warping and blending model that tackles anime talking head generation at multiple feature levels. It achieves smooth global transformation via low-level features and determines the rich details of anime characters via high-level features, as well as generating new content under large-displacement head movement with heavy occlusion.

Specifically, we first use a mask-guided generator to change the facial expression of source anime image. Then, we adopt an iterative grid generator to estimate accurate feature flow fields, which function as a guidance signal for head pose transformation. To tackle large-displacement head movement, the iterative grid generator generates flows step-by-step via feature iteration. Given estimated flows, instead of directly applying the flow-guided bilinear sampling in source anime image, we propose to apply sampling at feature levels, because the former can not tackle occlusion and may lead to non-smooth global transformation results. Therefore, we further design a hierarchical feature warping and blending generator that warps inputs at multiple feature levels, which enables the model to generate new content to tackle occlusion. In addition, we believe that the warped features of different levels contain different information. On the one hand, the low-level features can achieve smooth global transformation with poor details; on the other hand, the high-level features contain rich local details with non-smooth transformation. Thus we apply selective fusion units to blend these features, outputting final results with both smooth global transformation and rich local details. As shown in Fig. 1, our model can generate high-quality anime talking heads at 512×512 resolution using a single source anime image.

J. Zhang, C. Liu, K. Xian and Z. Cao are with the Key Laboratory of Image Processing and Intelligent Control, Ministry of Education; School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: jiale_zhang@hust.edu.cn; cx_liu@hust.edu.cn; kexian@hust.edu.cn; zgcao@hust.edu.cn).

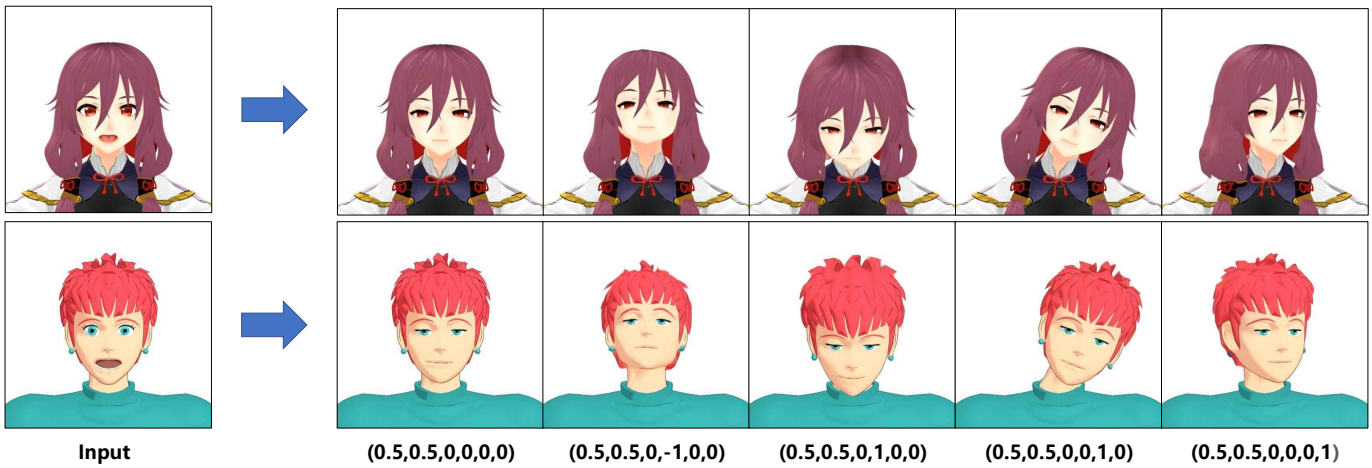

Figure 1. Our method can generate vivid high-quality anime talking heads at 512×512 resolution using only a single source anime image (the first column) and target pose vectors (the last row). Specifically, we encode the anime head pose into a pose vector (`left eye, right eye, mouth, top-down head, left-right head, yaw head`). The values for eyes and mouth range from 0 to 1, for head vary from -1 to 1.

Another important problem of this task is the shortage of available data. Intuitively, the dataset should contain a sufficient number of anime characters, where each character with varied poses. Since there is no such public anime pose dataset, we collect one termed AniTalk-2K. It contains 2046 anime IDs with 150 facial expression for each ID and 1980 anime IDs with 1108 head poses for each ID. All images in our AniTalk-2k are 512×512 resolution.

Extensive experiments demonstrate that our method can generate high-quality and photo-realistic anime talking heads at 512×512 resolution for arbitrary pose and outperforms prior state-of-the-art methods. Furthermore, experiments on AniTalk-2k show the effectiveness of our AniTalk-2K for high-quality anime talking head generation.

To sum up, the contributions of this work include:

- We propose a novel hierarchical feature warping and blending model for higher resolution and high-quality talking head animation;
- We create a high-quality anime talking head dataset AniTalk-2K that contains around 2000 anime characters with different face/head poses at 512×512 resolution;
- Experiments show that our model can generate vivid high-quality anime talking heads at 512×512 resolution and outperforms other state-of-the-art methods.

II. Related Work

A. Generative Adversarial Networks

GANs [1] have shown surprising results in various tasks since it was proposed, such as image generation [14], [15], image-to-image translation [16], [17], [18], human pose transformation [3], [19], [20], text-to-image translation [21], [22], [23], and face expression editing [10], [24]. Basic GANs include a generator and a discriminator, where the generator needs to generate real fake images and the discriminator aims to distinguish them. A based adversarial loss is used for the GAN model. Conditional GANs can generate condition-specific images by adding extra constraints to the basic GANs, such as text description [25], class information [26], and human pose information [19]. Thus they are applied to many tasks, *e.g.*, multi-domain transfer [27], human pose transformation [3], [28], [29], [30], and facial expression editing [10], [31].

B. Image-to-Image Translation

Image-to-image translation tasks [32], [17], [33], [34], [35] have also received much attention in recent years. For instance, some supervised methods like Pix2Pix [32] can achieve image-to-image translation using paired data. However, sometimes paired data may not be available in practice. To address this, some unsupervised methods are proposed [36], [17]. For instance, CycleGAN [36] learns a translation between source and target domains with unpaired data. It applies cycle-consistent loss that guarantees the consistency between generated images and original ones and maintains necessary attributes in input images. The method proposed by Gatys *et al.* [37] can transfer the inputs to be with other styles by using a content loss to preserve the content and a perceptual loss to convert the style to the target domain. CartoonGAN [17] can render the inputs of real world to anime style, while art2real [18] converts the artistic inputs to realistic images. Multi-domain image-to-image translation [38], [39], [40] is a different line where the same model can be applied to translate images to multi-domains using multiple conditions as input. For example, StarGAN [38] uses a single model to convert input images to multiple facial attributes changed results such as age, expression, and skin. In this paper, we focus on using paired data to achieve anime talking heads generation.

C. Appearance and Pose Transformation

In recent years, appearance and pose transformation has been extensively studied and made great progress.

Human. Most works focus on human image translation, such as facial attributes editing (*e.g.*, hair color and expression) [10], [38], human pose transformation [20], [41], and human talking heads generation [3], [13]. For example, GANimation [42] can

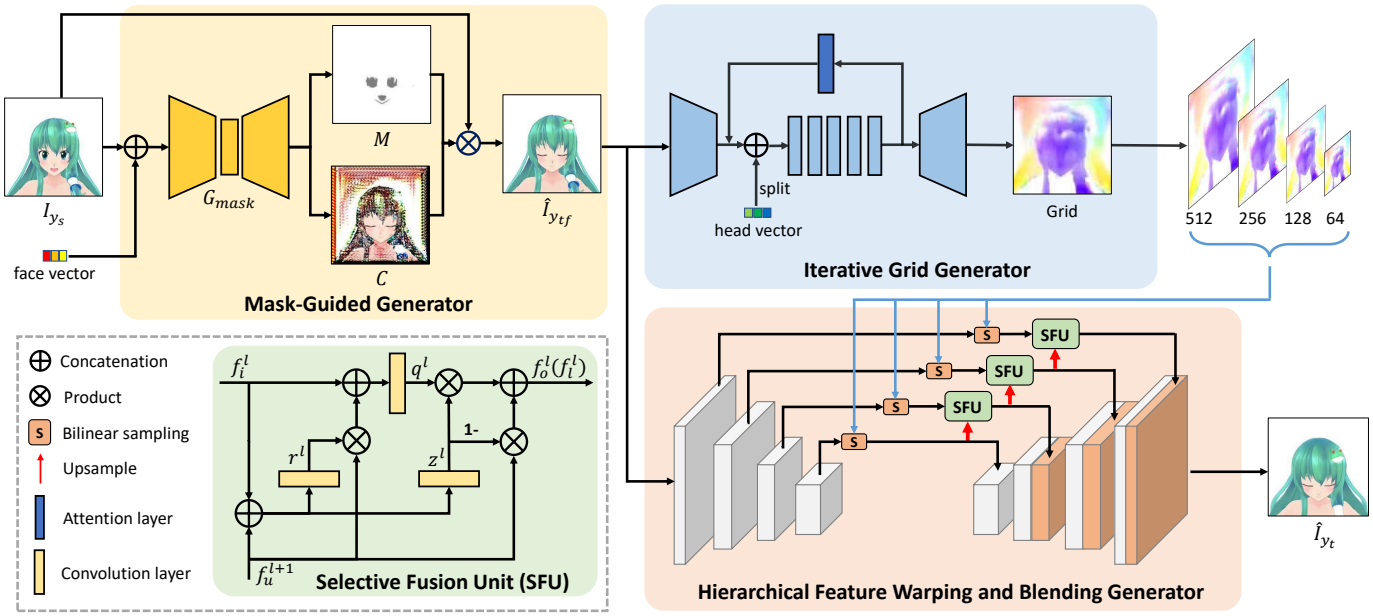

Figure 2. Overview of our HFWB. First a mask generator transforms an anime character to a target face pose (closed eyes and mouth), and then an iterative grid generator and a hierarchical feature warping and blending generator warp it to a target head pose. The bottom-left figure shows the details of SFU.

change the appearance and facial expression of human images. It requires a source human image and a target expression vector as input and then outputs a single channel mask and a content image. Finally, the input and content image are fused by linear combination through the mask to obtain the facial expression-changed human image conditioned on the target vector. For human pose transformation, Han *et al.* [29] propose a method that uses flow-based operation to implement body pose transformation of human. However, they warp the input directly and can not tackle occlusion. To address this, Ren *et al.* [30] propose a framework to first encode the body landmarks to latent space and then apply them to sample for features to achieve pose transformation. There are also many studies for human talking heads generation. For instance, given a target landmark, the few-shot learning method [3] can generate the target human pose based on a few source human images. The method [43] can re-create a talking-head video using only a single source image and a sequence of unsupervisedly-learned 3D keypoints. Based on audio, methods like [44] can also generate vivid human talking heads. However, these works are all designed for human and cannot tackle anime characters well.

Anime. There are also some works for anime talking head generation. For instance, MakeItTalk [13] can generate expressive talking-head videos from a single facial image, but it mainly uses physical models to process anime images. In addition, it tackles images at a low resolution of 256×256 and requires audio as input. The method proposed by Pramook [12] and CPTNet [11] can achieve anime pose transformation through a single anime image and a target vector. However, they still operate on 256×256 resolution, and can not tackle high resolution images with large displacement, heavy occlusion, and rich details.

D. Future Frame Prediction

Compared to our task of talking head animation, future frame prediction is a more generic problem, which can be used for abnormal event detection[45], video coding[46], and video completion[47]. There are several differences between these two tasks. For instance, future frame prediction can require a single frame[48] or a set of consecutive frames as input and generates the future frames, which does not require any annotation. Our task only requires a single image as input and generates the result that matches the target pose vector, which needs the annotated data. Also, without annotated data, the models of future frame prediction need to capture a notion of the complex dynamics of real-world phenomena to generate coherent sequences; while our model only needs to obtain the connection between the input image and target pose vector and generate a single result that has the appearance of origin anime as well as the target pose. Therefore, our task can actually be regarded as a specialized form of future frame prediction, including our own unique challenges as described in the introduction.

E. Diffusion Models

Diffusion models[49] have shown exciting performance for multiple tasks, including text-to-image generation and talking head animation [50], [51]. There are several differences between the diffusion model and our approach for this conditional generation task: i) cost of the training and inference. Diffusion model based approaches usually need more time and computing resources to synthesize a frame than other generative models because they have an iterative denoising process; ii) ability of conditional generation. Since the diffusion models have strong creativity, they may fail at preserving the original features of the inputs, especially for long sequences. In contrast, our approach uses an iterative

grid generator to sample the inputs, so we can maintain the input features well; iii) resolution of the data. Most of recent works for talking head using diffusion models are designed and trained on low-resolution human datasets, thus they may not work well on the high-resolution anime images with large pose transformation. In contrast, our model can achieve realistic high-resolution anime head animation even for large motion.

III. AniTalk-2K Dataset

We create an anime pose dataset—AniTalk-2K due to the lack of related public anime datasets for high resolution. The process of data creation and properties for AniTalk-2K are introduced in detail below.

A. Data Creation

We create our AniTalk-2K dataset with 3D software through three steps, including 3D model collection, 3D model processing, and pose annotation. The details are provided in the following.

a) 3D Models Collection: We first collect 3D animation models from different 3D model websites, such as BowlRoll [1] and Niconi solid [2]. Since there are various 3D models even for the tables or chairs in these websites, we manually filter and collect about 3000 3D models of anime characters in total. After removing duplicate and broken 3D models, we finally obtain 2180 qualified ones. These 3D models have rich 3D information of anime characters including 3D mesh and bones, thus we can obtain various poses for each anime character. For instance, the mesh can be used to control the appearances of 3D models, as well as the pose for components of face including eyes, eyebrow, and mouth (*e.g.*, close eyes and open mouth); the bones can control the movements of head (*e.g.*, roll, pitch, and yaw).

b) 3D Models Processing: In order to generate accurate and unified poses for each 3D anime model, it is necessary to use a 3D software named MikuMikuDance to process them first. Specifically, we design a series of facial expressions and head poses, and save them in an action file, which can guide each 3D anime character model into the same and unified poses. We also set the position of each anime head to the center of the image, and enlarge them so that the upper body of anime characters can occupy most of the space. Before rendering the model, we close the light and set the resolution of generated videos to 512×512, which is higher than previous methods [11], [12]. Then we break the generated videos into images. Note that the facial expression dataset and head pose dataset are created respectively, because we decouple this task into two stages, including facial expression changed stage and head pose transformed stage.

c) Pose Annotation: After processing the 3D models, we need to create annotation files to represent the facial expressions and head poses of each image. To achieve this, we first define the face pose vector, with the format of (`left eye, right eye, mouth`), to represent the facial expressions.

[1]https://bowlroll.net/

[2]https://3d.nicovideo.jp/

Table I. Comparison between our AniTalk-2K and other public anime datasets. 'C&P' means coarse and part; 'F&A' means fine and all.

Dataset	Image Nums	Anime Labels	Unified Style	Face Align.	Pose Anno.	Data Source
Danbooru 2019[52]	320K	✓	✗	✗	–	Media
Kaggle Anime[53]	63K	✗	✗	✗	–	Media
iCartoonFace[54]	0.4M	✓	✗	✗	C&P	Media
AniTalk-2K(Ours)	2.5M	✓	✓	✓	F&A	3D Models

The values of the face pose vector are from 0 to 1, where 0 means the eyes and mouth are fully closed and 1 means fully open. For instance, $(0, 1, 1)$ means that the anime character of the corresponding image has its left eye fully closed, right eye and mouth fully open. For the head pose dataset, we also define the head pose vector with the format of (`top-down head, left-right head, yaw head`) to represent the head poses. Different from the facial pose vector, the value range of head pose vector is $[-1, 1]$, which means the corresponding transformed angle of $[-20°, 20°]$. Note that some 3D anime models are not completely guided by action file due to the different settings used by the creators when making them. For instance, the left and right eyes of some 3D character models are always the same poses, which means we can not control one of them to open and the other to close at the same time. To address this, we carefully find these models and make the corresponding annotation files for them separately.

B. Dataset Properties

Our AniTalk-2K is a high quality flat dataset for 2D anime talking heads generation, which contains two subsets: a facial expression subset and a head pose subset. The facial expression subset includes 2046 anime IDs; each ID has 150 facial expression and corresponding pose vectors. Specifically, the eyes and mouth in face pose vectors have five values: [0, 0.25, 0.5, 0.75, 1]. As shown in Fig. 3, the face subset contains only eyes and mouth changed images. There are 1980 anime IDs for the head pose subset. Since head pose transformation is more difficult than changing facial expression, we use more postures for each anime ID to ensure AniTalk-2K contains a sufficiently comprehensive set of head poses. Specifically, each anime ID has 1108 head poses including yaw, pitch and roll and corresponding poses vectors. There are 9 values for each element in the head pose vector (`top-down head, left-right head, yaw head`): [−1, −0.75, −0.5, −0.25, 0, 0.25, 0.5, 0.75, 1]. The image resolution in our AniTalk-2K is 512×512, each anime character occupies the central area of the image and has rich details and colors. Also, all images in our AniTalk-2K are face aligned. Some examples are shown in Fig. 3.

C. Dataset Comparison

We make the comparison between our AniTalk-2K and other public animation head datasets, including the Danbooru 2019[52], Kaggle Anime Face[53], and iCartoonFace[54], The results are shown in Table I. There are three main advantages for our AniTalk-2K compared to these datasets. Firstly, since

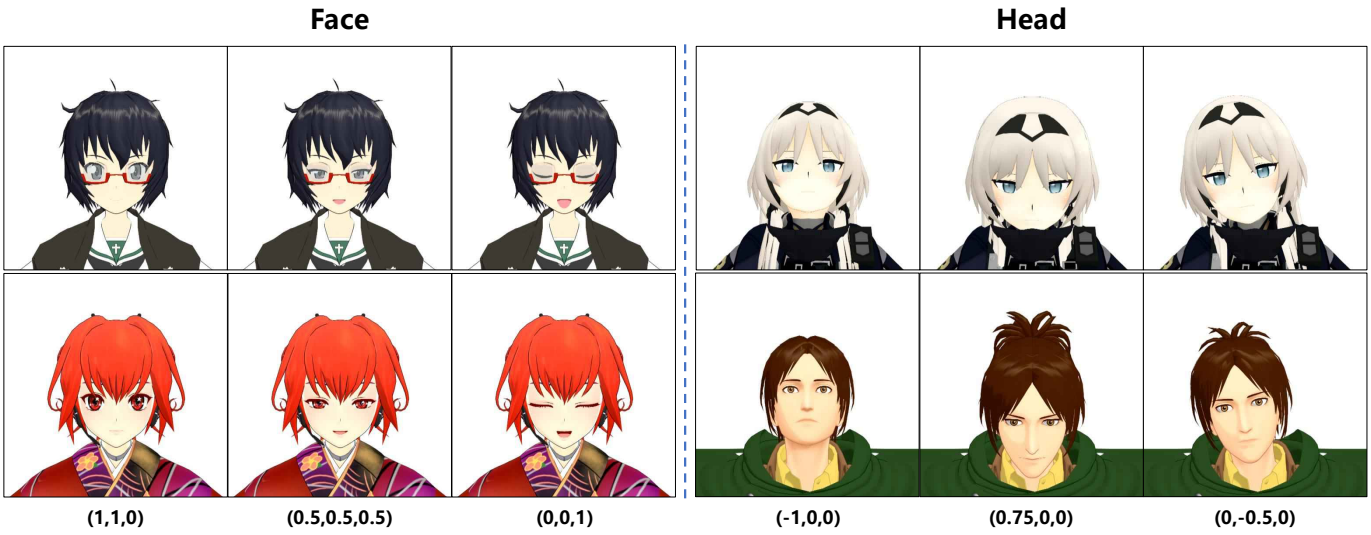

Figure 3. AniTalk-2K dataset. The left illustrates the anime facial expression subset and the right illustrates the anime head pose subset. The bottom row is the pose vector.

we create it with 3D models and 3D software while others are all from media, it is very standard and high quality with the high resolution of 512×512. Secondly, the facial expression and head pose are detailed annotated for each anime ID because we can finely control poses through the 3D models, while others either have no annotations or only have rough annotations for part of the data (Table I Attribute Anno.). Lastly, our AniTalk-2K provides more than $2.5M$ animation images with unified styles (Table I Image Nums, Unified Style). We believe these properties and advantages can facilitate the development of high-performance neural networks for a wide range of applications.

IV. Proposed Method

A. Overview

Our goal in this paper is to design a model to transform anime characters to arbitrary head poses at higher resolution (512×512).To this end, we propose a hierarchical feature warping and blending model (HFWB). As shown in Fig. 2, first a mask-guided generator is used to transform the anime character to a target facial expression, then an iterative grid generator and a hierarchical feature warping and blending generator are applied to generate the final head transformed anime images. To ease the training, we first disentangle this problem into two subtasks: facial expression transformation and head pose transformation. In face stage, the input only changes the eyes and mouth, so a simple mask-guided generator is used to maintain the other static areas. The outputs of the face stage are then used as the inputs of head stage, including the iterative grid generator and hierarchical feature warping and blending generator. In head stage, directly generating high resolution appearance flow is difficult, so we design a strategy of pose decomposition in the iterative grid generator. To tackle occlusion and fuse the information of different levels of features, the hierarchical feature warping and blending generator is proposed, in which we warp the features and use selective fusion units to achieve the combination.

a) Mask-Guided Generator: We apply a mask-guided generator G_{mask} to achieve the facial expression transformation as it is proved to be effective for human facial expression synthesis [42]. The details of G_{mask} are shown in Fig. 2. It requires an anime image I_{y_s} (512×512) and a target face pose vector y_t as input and outputs a single channel mask M and a content image C. Then the final output can be obtain by

$$\hat{I}_{y_{tf}} = M \cdot I_{y_s} + (1 - M) \cdot C \,, \tag{1}$$

where $\cdot$ denotes element-wise multiplication. The mask M can guide the network to focus on areas that need to be changed like the eyes and mouth. The static areas can be directly obtained from the input image, and other transformed areas are generated through the content image C. This eases the learning process as the network only needs to focus on generating dynamic areas. However, for head pose transformation, G_{mask} can easily generate artifacts and blur due to the large displacement and heavy occlusion.

B. Iterative Grid Generator

Since the mask generator G_{mask} can not tackle head pose transformation, we adopt a grid-based generator G_{grid} with feature iteration to transform the head poses of anime images. The details are shown in Fig. 2. It receives a facial expression changed image $\hat{I}_{y_{tf}}$ (512×512) and a target head pose vector as input, and outputs a two-channel grid (512×512). Similar to appearance flow [55], the grid is used to warp the image or features to target poses by bilinear sampling since it can preserve the pixel information of the source images or features. However, due to the limited receptive field of Convolution Neural Networks, it is difficult to directly generate high resolution grid for large displacement. To address this, we propose a strategy of pose decomposition and feature iteration. Specifically, we decompose the large head pose vector into k small pose vectors and perform k iterations for the residual blocks. For instance, if the value of head pose vector is

$(0, 0, 1)$, we decompose it into k vectors with the value of $(0, 0, 1/k)$. Then we add each of them to the input of residual blocks for every feature iteration. Furthermore, we apply an attention layer for each iteration, which generates a mask feature and a content feature to update the output of residual layers. In practice, we observe that $k = 4$ is sufficient to tackle this problem. This operation enables the model to generate high resolution flow field step by step like human.

However, directly applying the grid to high resolution source anime images may cause many artifacts when it comes to large displacement, intricate details, and heavy occlusion due to lack of ability to create new pixel.

C. Hierarchical Feature Warping and Blending Generator

To address the problem mentioned above, we propose the hierarchical feature warping and blending Generator. The details are shown in Fig. 2. It receives the facial expression changed anime image $\hat{I}_{y_{tf}}$ (512×512) as input, outputting transformed image $\hat{I}_{y_t}$. Since directly warping the high-resolution input anime images may generate artifacts, we propose to warp input data at multiple feature levels. Specifically, we first downsample the generated 512×512 grid to 256×256, 128×128, and 64×64 resolutions, and then apply them to warp the multiple encoder features of the corresponding resolution with bilinear sampling. This operation enables the network to generate new content to tackle heavy occlusion. Further, we observe that low-resoluition grids can achieve better global head pose transformation than high, while high-resolution grids can retain richer and finer local details than low. Thus we design the selective fusion units that fuse the warped features for multiple resolutions level-by-level. After fusion, the multiple features are skip connected to the corresponding decoder features to obtain final results with both smooth global transformation and rich local details.

Selective Fusion Units. We propose selective fusion units (SFUs) to selectively fuse warped encoder features of different resolutions and transfer to decoder features via skip connection. Fig. 2 illustrates the details. We choose to modify the GRU [56] to design our SFU due to the need for selection and fusion.

We demonstrate the process by taking l-th encoder layer as an example. Let ${f^l}_i$ denote the encoder feature of l-th layer, ${f^l}_o$ denote the the fusion feature of l-th layer, and ${f^{l+1}}_l$ denote the low-resolution feature of $(l+1)$-th layer. First, we upsample the ${f^{l+1}}_l$ to obtain the ${f^{l+1}}_u$. Then we fuse them by

$$r^l = Conv1\left(\left[{f^l}_i, {f^{l+1}}_u\right]\right), \tag{2}$$

$$z^l = Conv2\left(\left[{f^l}_i, {f^{l+1}}_u\right]\right), \tag{3}$$

$$q^l = Conv3\left(\left[{f^{l+1}}_u \cdot r^l, {f^l}_i\right]\right), \tag{4}$$

$${f^l}_o = z^l \cdot q^l + \left(1 - z^l\right) \cdot {f^{l+1}}_u, \tag{5}$$

where $Conv1$, $Conv2$, and $Conv3$ denote the convolution layers, $\cdot$ denotes element-wise multiplication.

On the one hand, ${f^l}_o$ denotes the output of l-th SFU. On the other hand, it functions as the low-resolution feature of l-th layer $({f^l}_l)$. By resetting r^l and updating z^l, we can choose which features to fuse and which to discard. Further, we can control the fusion weights of features via q^l and linear combination. This enables our model to take the advantages of warped features from multiple levels, leading to better global transformation and finer maintenance of local details.

D. Loss Function

We define four loss terms in total at the training stage:

- the content loss for constraining the consistency of generated images and ground truth ones;
- the perceptual loss used to enhance the details and sharpness;
- the adversarial loss that improves the realism of generated results;
- the offset loss to limit the range of bilinear sampling by grids.

Since our AniTalk-2K dataset contains paired samples, we apply ℓ_1 loss to constrain the generator to generate results which are consistent with ground truth images. The content loss is defined by

$$\mathcal{L}_{con} = E_{I_{y_t} \sim P_{data}}\left[\left\| I_{y_t} - \hat{I}_{y_t} \right\|_1\right], \tag{6}$$

where $\hat{I}_{y_t}$ is the final head pose transformed results, the I_{y_t} is the corresponding ground truth images, and P_{data} denotes the distribution of real images.

However, solely relying on ℓ_1 loss to optimize the model may lead to blurred results. Therefore, we further apply perceptual loss [57] as another constraint. Specifically, the generated results and its corresponding ground truth images are sent to the pre-trained $VGG19$ model [58], then we extract the features from the $conv1\_1$, $conv2\_1$, and $conv3\_1$ layers to compute the perceptual loss, which is given by

$$\mathcal{L}_p = \sum_j E_{I_{y_t} \sim P_{data}}\left[\left\| \phi_j\left(I_{y_t}\right) - \phi_j\left(\hat{I}_{y_t}\right) \right\|_1\right], \tag{7}$$

where $\phi_j(\cdot)$ represents the $VGG19$ model features of the jth layer.

Since the adversarial loss is proved effective to improve the realism of generations, we adopt it in our model, which is defined by

$$\mathcal{L}_{adv} = E_{\hat{I}_{y_t} \sim P_s}\left[D\left(\hat{I}_{y_t}\right)\right] - E_{I_{y_t} \sim P_{data}}\left[D\left(I_{y_t}\right)\right], \tag{8}$$

where the P_s stands for the data distribution of generated images. In the adversarial process, the generator aims at generating real-fake images to maximize the objective, while the discriminator D aims to distinguish between real and fake images to minimize the objective.

We observe that it is difficult to generate accurate grid without constraint, because of the large search space. Thus we propose offset loss to limit the sampling range of the grid. It is defined by

$$\mathcal{L}_{off} = \left\| g_t - g_i \right\|_1, \tag{9}$$

Figure 4. Pose transformation results. The top-left image with red rectangle is the input, and the rest are all generated results.

where the g_t represents the grid generated by G_{grid}, and g_i is the preset grid that guides the source image to transform to the unchanged image after bilinear sampling. We observe this loss can improve the generated grid and accelerate convergence.

The final loss is defined as

$$\mathcal{L} = \mathcal{L}_{adv} + \lambda_1 \mathcal{L}_{con} + \lambda_2 \mathcal{L}_p + \lambda_3 \mathcal{L}_{off} \,, \tag{10}$$

where λ_1, λ_2, and λ_3 are hyper-parameters.

Table II. Quantitative comparisons on AniTalk-2K with state-of-the-art methods including StarGAN [38], Khungurn's method [12], CPTNet [11], and Ren *et al.* [59].

Method	MAE ↓	RMSE ↓	PSNR ↑	SSIM ↑	FVD ↓
StarGAN [38]	0.064	0.191	21.060	0.812	219.6
Khungurn's [12]	0.054	0.182	24.740	0.833	151.8
CPTNet [11]	0.048	0.167	25.179	0.841	138.3
Ren *et al.* [59]	0.049	0.170	24.917	0.844	144.8
Ours	**0.040**	**0.156**	**25.934**	**0.861**	**122.4**

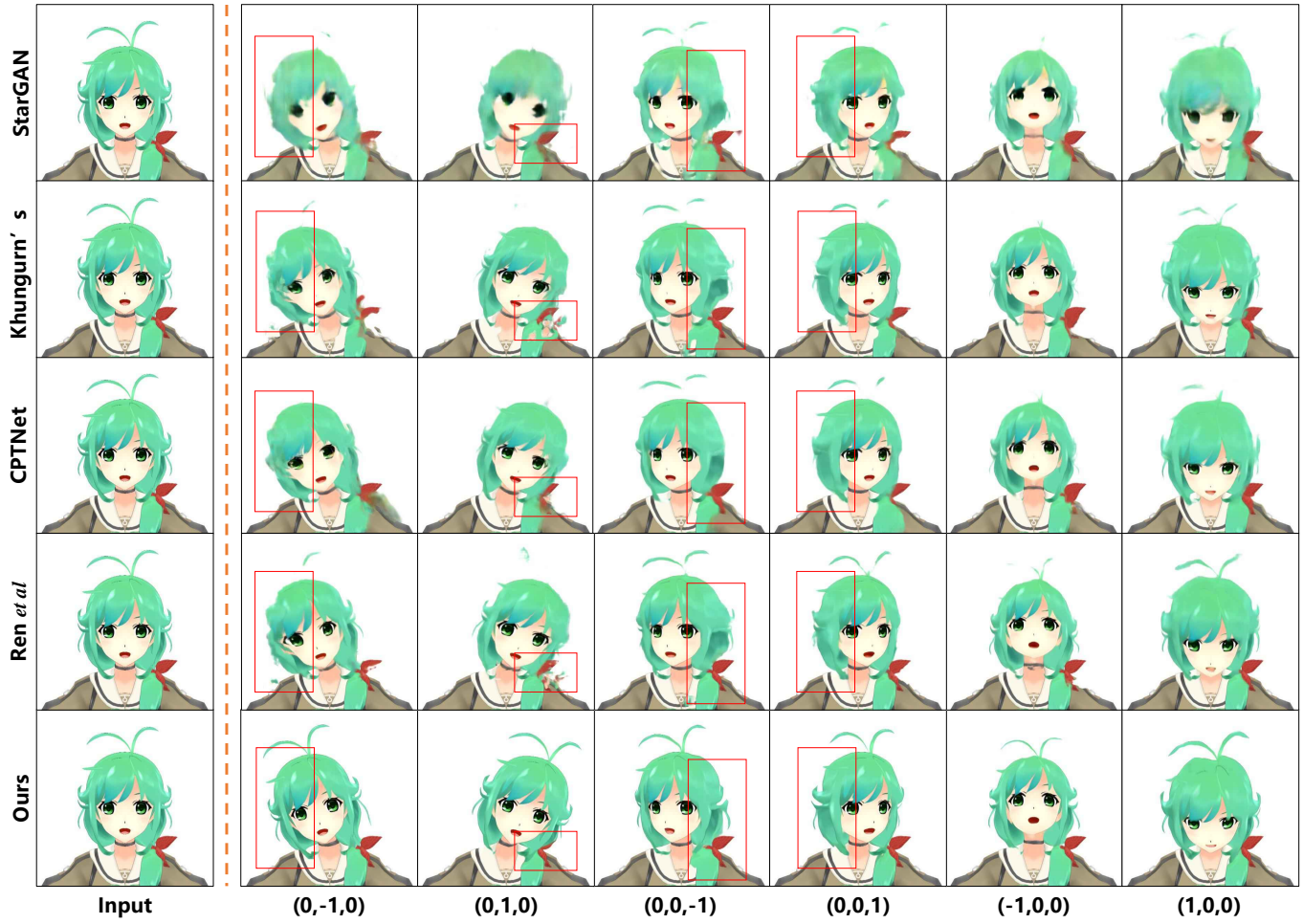

Figure 5. Qualitative comparison with several state-of-the-art models including StarGAN [38], Khungurn's method [12], CPTNet [11], and Ren *et al.* [59]. We select seven pose vectors for comparisons, and the 2^{nd}–8^{th} columns are the generated images.

Table III. Quantitative comparisons on AnimeCeleb[60] with state-of-the-art methods.

Method	MAE ↓	RMSE ↓	PSNR ↑	SSIM ↑	FVD ↓
StarGAN [38]	0.066	0.196	17.81	0.776	286.1
Khungurn's [12]	0.059	0.188	19.89	0.798	230.8
CPTNet [11]	0.053	0.181	22.01	0.811	207.5
Ren *et al.* [59]	0.055	0.179	21.82	0.815	198.6
Ours	**0.048**	**0.171**	**23.15**	**0.828**	**171.8**

V. Results and Discussions

A. Implementation Details

We train the model on the AniTalk-2K dataset. We use Adam [61] to optimize the generator with a learning rate of 0.0001, $\beta_1 = 0.5$, $\beta_2 = 0.999$, and a batch size of 6. All networks are trained from scratch and we train the mask-guided generator with $200,000$ iterations, iterative grid generator with $800,000$ iterations, and hierarchical feature warping and blending generator with $1,000,000$ iterations, respectively. We set the model hyper-parameters $\lambda_1 = 1000$, $\lambda_2 = 200$, and $\lambda_3 = 10$. We train our model with 6 NVIDIA GeForce RTX 3090 GPUs. It takes one day to train our mask-guided generator, three days for the iterative grid generator, and four days for the hierarchical feature warping and blending generator. Note that we train and test our model at resolution of 512×512.

We use five metrics including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Peak Signal-to-Noise Ratios (PSNR), Structural Similarity (SSIM), and Fréchet Video Distance (FVD)[62] for evaluation. MAE and RMSE are used to measure the average errors between ground truth and the generated image, PSNR is adopted to measure the difference between generated results and ground truth based on the maximum pixel value and Mean Square Errors (MSE), SSIM quantifies the perceptual similarity between them, and FVD measures overall visual quality and temporal coherence without reference to the ground truth.

B. Quantitative Comparison

Here we compare our approach with several state-of-the-art methods, including StarGAN [38], Khungurn [12], CPTNet [11], and Ren *et al.* [59]. We evaluate these methods on the testing set of AniTalk-2K, where each anime ID has 27 different poses. Quantitative comparison results are shown in Table II, where lower MAE, RMSE, and FVD scores and higher PSNR and SSIM scores mean better generations. These

Figure 6. Qualitative comparison with several state-of-the-art models on AnimeCeleb[60]. We select eight pose vectors for comparisons, and the 2^{nd}–9^{th} columns are the generated images.

methods for comparison are all trained using our dataset at 512×512 resolution. StarGAN achieves pose transformation by directly outputting results through convolutional layers, which leads to poor performance. In contrast to CPTNet and the approach of Khungurn that use the flows to sample the inputs directly, the method proposed by Ren *et al.* samples the features so that it can obtain better performance. Since our model generates accurate high-resolution flows by feature iteration and uses low-level features to control global transformation and high-level features to determine the local details, we achieve best MAE, RMSE, PSNR, and SSIM among all methods of comparison, which validates the effectiveness of our approach.Furthermore, our method achieves lowest FVD, which demonstrates our model can generate more realistic, accurate, and consecutive results. Note that since there is a large percentage of white backgrounds in the images for our AniTalk-2K, the gap between our approach and other methods in four metrics is narrowed.

To further validate the effectiveness and improvement of our approach, we conduct a comparison with baselines on other benchmarks. Since there is no related high resolution anime pose dataset, we choose AnimeCeleb[60], which is similar to our data. The resolution of it is 256×256, and the results of comparison are shown in Table III. Though our approach is designed for high resolution anime images, we can still achieve best scores for the all five metrics on the AnimeCeleb, which further shows the superiority of our model.

We also provide the comparison of single frame inference speed with other methods. The results are shown in Table VI. StarGAN uses a naive generator to directly output results, so it achieves the fastest inference speed. Our approach can obtain better results while spending less inference time than other methods, which demonstrates the effectiveness of our model.

C. Qualitative Comparison

Qualitative results are shown in Fig. 5. StarGAN tends to generate results with many artifacts, and is not able to obtain clear results at high resolution, especially when it comes to large-displacement pose transformation. Khungurn's method and CPTNet obtain relatively sharper results than StarGAN, however, they can not achieve smooth global transformation such as the 3^{th} column and 5^{th} column corresponding to the pose vectors of $(0, 1, 0)$ and $(0, -1, 0)$. The reasons are that they are all designed for low resolution anime images at 256×256 resolution, when it comes to high resolution (512×512) with large displacement and rich details, they struggle to generate clear results. The method proposed by Ren *et al.* is able to generate clearer results than the other three models mentioned above, but it can not keep the details such as decoration, hair, and clothes. As shown in the red rectangle, the blurs and artifacts appear obviously.

Instead, our approach can obtain clearer and sharper results with less blur and artifacts than other models. As shown in the last row, because we apply hierarchical feature warping and blending for our method, the results of our model are able to not only achieve smooth global pose transformation, but also maintain the intricate textures and rich details with high fidelity, such as hair, decoration, and eyes. Further, since we use the iterative grid generator to generate high resolution

Figure 7. Ablation study of our method. Our full model can obtain both smooth global transformation and fine local details.

flows step-by-step and warp the inputs at the feature level, our approach is capable of generating new content to tackle heavy occlusion caused by large displacement.

Qualitative comparison on the AnimeCeleb dataset is shown in Fig. 6. Similar to the results on our AniTalk-2K, our approach can obtain sharp and realistic outputs while other methods tend to generate blur and artifacts especially for the large motion. Because our model can iteratively generate accurate flows and uses the strategy of hierarchical feature warping and blending to combine both global and local information.

We also provide additional pose transformation results with the change of both facial expression and continuous head poses. As shown in Fig. 4, given a single anime image (with red border) as input, our model can generate various images with mixed poses, which are clear and sharp with both smooth global transformation and rich local details thanks to the hierarchical structure of our approach.

D. Ablation Study

In this section, we conduct ablation study to evaluate each component of our model. Specifically, we compare our model with the following variants:

- *Baseline*. We use an auto-encoder network as the baseline, which requires a source anime image and a target pose as input, and outputs the pose-changed results directly.
- *Baseline-Up*. We adopt a grid generator to generate flows directly at resolution of 128×128 and then upsample it to 512×512. Then we use the flows to sample the source anime inputs to obtain results at 512×512 resolution.
- *Baseline-Grid*. We use the grid generator to generate flows at 512×512 resolution directly and then sample the anime inputs to obtain results.
- *Grid-Iter*. We use the same network architecture as the iterative grid generator in our model without the hierarchical feature warping and blending generator, and apply the flows generated to sample the source inputs directly to generate results.

Figure 8. Visualization of the flows. We visualize the appearance flow grid and corresponding images specific to the pose vector. Different colors represent different moving directions according to the color wheel and the magnitude of color indicates the displacement in the flow.

Table IV. Quantitative results of different k in Iterative Grid Generator.

Method	MAE ↓	RMSE ↓	PSNR ↑	SSIM ↑	FVD ↓
$k=1$	0.058	0.196	22.182	0.828	180.3
$k=2$	0.052	0.179	23.863	0.839	163.9
$k=3$	0.049	0.171	24.562	0.845	155.8
$k=4$	**0.047**	**0.168**	**24.915**	**0.848**	**151.6**

- *Full-Model*. Our full model is used.

We adopt the same metrics mentioned above to evaluate our model. The quantitative results are shown in Table V, where lower MAE, RMSE scores and higher PSNR, SSIM scores imply better results. The qualitative results of different models are shown in Fig. 7. One can observe that 'Baseline' obtains the worst results and generates heavy artifacts and blurs. This indicates that naive encoder-decoder architecture can not tackle large pose transformation at high resolution directly.

Efficacy of Iterative Grid Generator. Here we evaluate the effectiveness of our iterative grid by comparing 'Grid-Iter' with 'Baseline-Up' and 'Baseline-Grid'. As shown in Table V, 'Baseline-Up' and 'Baseline-Grid' can achieve better performance than 'Baseline', which indicates that the flow-based method is an effective way to tackle spatial pose transformation. However, we observe that 'Baseline-Grid' suffers difficulty in generating high resolution flows under large-displacement head movement, as illustrated in Fig. 7. Thanks to the iterative design of our grid generator, *i.e.*, generating flows step-by-step via feature iteration, 'Grid-Iter' can achieve better performance than 'Baseline-Grid'. Interestingly, we also observe that 'Baseline-Up' achieves more smooth transformation than 'Baseline-Grid'. The reason is that 'Baseline-Up' generates flows at low resolution and upsamples it to high resolution. However, its details are not satisfactory. On the contrary, 'Grid-Iter' generates high resolution flows directly, thus it can obtain richer details than 'Baseline-Up'. The results in Table V and Fig. 7 both validate the effectiveness of our iterative grid generator.

We also visualize the two-channel flow output in Fig. 8. We provide generated flows of our iterative grid generator and the target anime results of sampling for each pose vector. Our model can generate smooth and accurate flows according to the color wheel. Note that the colors represent the direction of motion and the magnitude of colors indicates the displacement (the darker the color is, the larger displacement occurs).

To further explore the influence of pose decomposition and feature iteration strategy, we provide the quantitative results of different k in Table IV. It can be observed that the performance of iterative grid generator is improved when the k is increased, but the range is decreased. So we finally choose $k=4$ in our model considering the effectiveness and efficiency.

Efficacy of HFWB generator. We evaluate the effectiveness of our HFWB generator by comparing 'Full-Model' with 'Grid-Iter'. Although iterative grid generator can generate more accurate and smooth flows than baselines, it can not tackle intricate details and heavy occlusion by sampling the source input directly. As shown in Fig. 7, unnatural textures and artifacts appear in the red rectangles. In contrast, our HFWB generator samples the features to transform inputs, which can generate new content to tackle occlusion. As shown in the red rectangles, our full model can obtain more smooth global transformation and better maintain for rich details than 'Grid-Iter' since our HFWB generator can fuse information of multi-levels features.

Overall, our 'Full-Model' obtains the best results in both quantitative and qualitative evaluation, which demonstrates the efficacy of our approach.

Figure 9. Application of our method for live action following. 'Target' denotes the human driving image, and 'Output' is the anime generating result.

Table V. Evaluation results of Ablation Study on AniTalk-2K.

Method	MAE ↓	RMSE ↓	PSNR ↑	SSIM ↑	FVD ↓
Baseline	0.061	0.190	21.527	0.821	197.4
Baseline-Up	0.053	0.177	22.327	0.835	158.3
Baseline-Grid	0.058	0.196	22.182	0.828	180.3
Grid-Iter	0.047	0.168	24.915	0.848	151.6
Full-Model	**0.040**	**0.156**	**25.934**	**0.861**	**122.4**

Table VI. Comparison of single frame inference speed and some quantitative results with other methods.

Method	Time ↓	PSNR ↑	SSIM ↑	FVD ↓
StarGAN[38]	**0.006s**	21.060	0.812	219.6
Khungurn's[12]	0.068s	24.740	0.833	151.8
CPTNet[11]	0.161s	25.179	0.841	138.3
Ren *et al.* [59]	0.049s	24.917	0.844	144.8
Ours	0.044s	**25.934**	**0.861**	**122.4**

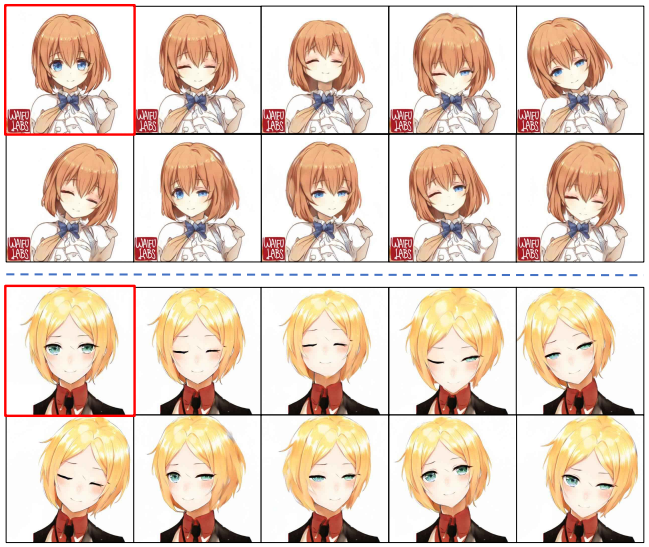

Figure 10. Results of out-of-dataset images. The input images are not included in our dataset and downloaded from the Internet.

E. Generalization

In this section, we evaluate the generalization of our model with out-of-dataset anime images. As shown in Fig. 10, the two anime images with red borders are the inputs, which are generated and downloaded by WaifuLabs.[3] The rest of the images are all generated by our model. For various pose transformation including the change of facial expression and the movement of anime heads, our approach can generate clear and sharp results with both smooth global transformation and rich local details. This indicates the good generalization of our method. Note that the model is trained on our AniTalk-2K dataset without extra data.

[3]https://waifulabs.com/

VI. Application

We also demonstrate an interesting application of our model. Given a human image as a driven image and an anime image as input, we can generate the pose-transformed anime image following the human images including facial expression and head motion. Specifically, we first obtain the landmark of human face and then convert it into a pose vector, finally the pose vector is passed into our model to guide the anime image to generate pose-transformed results. As shown in Fig. 9, our model can generate clear results and accurately follow the human.

VII. Conclusion

In this paper, we study talking head animation from a single anime image at a higher resolution, *i.e.*, 512×512. To address the shortage of data, we first present an anime pose dataset—AniTalk-2K for benchmarking high-resolution talking head animation. Due to its high-quality, we believe this dataset can help advance the progress of this task. Further, we propose a Hierarchical Feature Warping and Blending (HFWB) model to tackle high-resolution anime talking-head generation. The core idea of our approach is to transform anime pose hierarchically at feature levels, *i.e.*, it achieves smooth global transformation based on low-level features while maintaining rich local details via high-level features, as well as generating new content to tackle occlusion. Extensive experiments justify that our approach enables high-resolution and high-quality talking head animation, outperforming prior state-of-the-art methods.

References

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proceedings of the Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[2] T.-C. Wang, A. Mallya, and M.-Y. Liu, "One-shot free-view neural talking-head synthesis for video conferencing," *arXiv preprint arXiv:2011.15126*, 2020.

[3] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky, "Few-shot adversarial learning of realistic neural talking head models," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9459–9468.

[4] O. Wiles, A. Sophia Koepke, and A. Zisserman, "X2face: A network for controlling face generation using images, audio, and pose codes," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 670–686.

[5] F.-T. Hong, L. Zhang, L. Shen, and D. Xu, "Depth-aware generative adversarial network for talking head video generation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2022, pp. 3397–3406.

[6] Y. Ren, G. Li, Y. Chen, T. H. Li, and S. Liu, "Pirenderer: Controllable portrait image generation via semantic neural rendering," in *Proceedings of the IEEE international Conference on computer vision*, 2021, pp. 13 759–13 768.

[7] L. Chen, G. Cui, C. Liu, Z. Li, Z. Kou, Y. Xu, and C. Xu, "Talking-head generation with rhythmic head motion," in *Proceedings of the European Conference on Computer Vision*. Springer, 2020, pp. 35–51.

[8] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, "Neural voice puppetry: Audio-driven facial reenactment," in *Proceedings of the European Conference on Computer Vision*. Springer, 2020, pp. 716–731.

[9] M. Liu, Y. Ding, M. Xia, X. Liu, E. Ding, W. Zuo, and S. Wen, "Stgan: A unified selective transfer network for arbitrary image attribute editing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3673–3682.

[10] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, "Attgan: Facial attribute editing by only changing what you want," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5464–5478, 2019.

[11] J. Zhang, K. Xian, C. Liu, Y. Chen, Z. Cao, and W. Zhong, "Cptnet: Cascade pose transform network for single image talking head animation," in *Asian Conference on Computer Vision*, 2020.

[12] P. Khungurn, "Talking head anime from a single image," https://pkhungurn.github.io/talking-head-anime/, 2019.

[13] Y. Zhou, X. Han, E. Shechtman, J. Echevarria, E. Kalogerakis, and D. Li, "Makelttalk: speaker-aware talking-head animation," *ACM Transactions on Graphics*, vol. 39, no. 6, pp. 1–15, 2020.

[14] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410.

[15] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8110–8119.

[16] Y. Zhao, R. Wu, and H. Dong, "Unpaired image-to-image translation using adversarial consistency loss," *arXiv preprint arXiv:2003.04858*, 2020.

[17] Y. Chen, Y.-K. Lai, and Y.-J. Liu, "Cartoongan: Generative adversarial networks for photo cartoonization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9465–9474.

[18] M. Tomei, M. Cornia, L. Baraldi, and R. Cucchiara, "Art2real: unfolding the reality of artworks via semantically-aware image-to-image translation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5849–5859.

[19] 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*, 2019, pp. 5904–5913.

[20] F. Ma, G. Xia, and Q. Liu, "Spatial consistency constrained gan for human motion transfer," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[21] M. Yuan and Y. Peng, "Bridge-gan: interpretable representation learning for text-to-image synthesis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 4258–4268, 2019.

[22] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "Attngan: Fine-grained text to image generation with attentional generative adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1316–1324.

[23] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5907–5915.

[24] Y. Xia, W. Zheng, Y. Wang, H. Yu, J. Dong, and F.-Y. Wang, "Local and global perception generative adversarial network for facial expression synthesis," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[25] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," *arXiv preprint arXiv:1605.05396*, 2016.

[26] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 2642–2651.

[27] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "Stargan v2: Diverse image synthesis for multiple domains," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8188–8197.

[28] C. Chan, S. Ginosar, T. Zhou, and A. A. Efros, "Everybody dance now," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5933–5942.

[29] X. Han, X. Hu, W. Huang, and M. R. Scott, "Clothflow: A flow-based model for clothed person generation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 10 471–10 480.

[30] Y. Ren, G. Li, S. Liu, and T. H. Li, "Deep spatial transformation for pose-guided person image generation and animation," *IEEE Transactions on Image Processing*, vol. 29, pp. 8622–8635, 2020.

[31] R. Wu, G. Zhang, S. Lu, and T. Chen, "Cascade ef-gan: Progressive facial expression editing with local focuses," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5021–5030.

[32] 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*, 2017, pp. 1125–1134.

[33] H. Emami, M. M. Aliabadi, M. Dong, and R. Chinnam, "Spa-gan: Spatial attention gan for image-to-image translation," *IEEE Transactions on Multimedia*, 2020.

[34] H. Zhang, V. Sindagi, and V. M. Patel, "Image de-raining using a conditional generative adversarial network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 3943–3956, 2019.

[35] D. S. Tan, Y.-X. Lin, and K.-L. Hua, "Incremental learning of multi-domain image-to-image translations," *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.

[36] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2223–2232.

[37] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2414–2423.

[38] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8789–8797.
[39] A. Anoosheh, E. Agustsson, R. Timofte, and L. Van Gool, "Combogan: Unrestrained scalability for image domain translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 783–790.
[40] X. Yang, D. Xie, and X. Wang, "Crossing-domain generative adversarial networks for unsupervised multi-domain image-to-image translation," in *Proceedings of the 26th ACM international conference on Multimedia*, 2018, pp. 374–382.
[41] Y. Li, C. Huang, and C. C. Loy, "Dense intrinsic appearance flow for human pose transfer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3693–3702.
[42] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: Anatomically-aware facial animation from a single image," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 818–833.
[43] T.-C. Wang, A. Mallya, and M.-Y. Liu, "One-shot free-view neural talking-head synthesis for video conferencing," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2021, pp. 10 039–10 049.
[44] D. Zeng, H. Liu, H. Lin, and S. Ge, "Talking face generation with expression-tailored generative adversarial network," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 1716–1724.
[45] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection–a new baseline," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6536–6545.
[46] W. Lotter, G. Kreiman, and D. Cox, "Deep predictive coding networks for video prediction and unsupervised learning," *arXiv preprint arXiv:1605.08104*, 2016.
[47] C. Finn, I. Goodfellow, and S. Levine, "Unsupervised learning for physical interaction through video prediction," *Advances in neural information processing systems*, vol. 29, 2016.
[48] L. Yu, Y. Cheng, K. Sohn, J. Lezama, H. Zhang, H. Chang, A. G. Hauptmann, M.-H. Yang, Y. Hao, I. Essa *et al.*, "Magvit: Masked generative video transformer," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2023, pp. 10 459–10 469.
[49] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.
[50] C. Xu, S. Zhu, J. Zhu, T. Huang, J. Zhang, Y. Tai, and Y. Liu, "Multimodal-driven talking face generation, face swapping, diffusion model," *arXiv preprint arXiv:2305.02594*, 2023.
[51] S. Shen, W. Zhao, Z. Meng, W. Li, Z. Zhu, J. Zhou, and J. Lu, "Difftalk: Crafting diffusion models for generalized audio-driven portraits animation," in *Proceedings of the IEEE Conference on computer vision and pattern recognition*, 2023, pp. 1982–1991.
[52] D. C. Gwern Branwen, Anonymous, "A large-scale anime character illustration dataset," https://www.gwern.net/Crops, 2020.
[53] "Kaggle animation face," https://www.kaggle.com/splcher/animefacedataset.
[54] Y. Zheng, Y. Zhao, M. Ren, H. Yan, X. Lu, J. Liu, and J. Li, "Cartoon face recognition: A benchmark dataset," in *Proceedings of the 28th ACM international conference on multimedia*, 2020, pp. 2264–2272.
[55] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros, "View synthesis by appearance flow," in *Proceedings of the European Conference on Computer Vision*. Springer, 2016, pp. 286–301.
[56] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
[57] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proceedings of the European Conference on Computer Vision*. Springer, 2016, pp. 694–711.
[58] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[59] J. Ren, M. Chai, O. J. Woodford, K. Olszewski, and S. Tulyakov, "Flow guided transformable bottleneck networks for motion retargeting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 795–10 805.
[60] K. Kim, S. Park, J. Lee, S. Chung, J. Lee, and J. Choo, "Animeceleb: Large-scale animation celebheads dataset for head reenactment," in *Proceedings of the European Conference on Computer Vision*, 2022, pp. 414–430.
[61] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[62] T. Unterthiner, S. Van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly, "Towards accurate generative models of video: A new metric & challenges," *arXiv preprint arXiv:1812.01717*, 2018.

Jiale Zhang received the B.S. and M.S. degree from Huazhong University of Science and Technology, Wuhan, China.

His research interests include computer vision and machine learning, with a particular emphasis on image-to-image translation, generative adversarial networks, and neural rendering for autonomous driving.

Chengxin Liu received the B.S. degree from Huazhong University of Science and Technology, Wuhan, China, in 2018, where he is currently pursuing the Ph.D. degree with the School of Artificial Intelligence and Automation.

His research interests include computer vision and machine learning, with a particular emphasis on label noise learning and its applications to dense prediction tasks, such as object detection, object counting, and segmentation.

Ke Xian received the B.S. and Ph.D. degrees from Huazhong University of Science and Technology, Wuhan, China. From 2021 to 2023, he was a Research Fellow with the S-Lab, Nanyang Technological University, Singapore. Now, he is a lecturer with the School of Electronic Information and Communications, Huazhong University of Science and Technology.

His research interests include computer vision and computational photography with a focus on robust depth estimation, and neural generation/rendering/editing.

Zhiguo Cao (Member, IEEE) received the B.S. and M.S. degrees in communication and information system from the University of Electronic Science and Technology of China, Chengdu, China, and the Ph.D. degree in pattern recognition and intelligent system from Huazhong University of Science and Technology, Wuhan, China. He is currently a Professor with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology.

His research interests include computational photography, monocular depth estimation, 3D video processing, motion detection, and human action analysis. He has authored and coauthored dozens of papers in international journals and prominent conferences, which have been applied to image processing in mobile phone cameras and automatic observation systems for crop growth in agriculture and for weather phenomena in meteorology.