AFRL-AFOSR-JP-TR-2023-0064

Knowledge Externalization from Image-to-Image Translation

Jaegul Choo
Korea Advanced Institute of Science and Technology
291 Daehak-ro, Yuseong-gu
Taejon, ,
KR

03/02/2023
Final Technical Report

Air Force Research Laboratory
Air Force Office of Scientific Research
Asian Office of Aerospace Research and Development
Unit 45002, APO AP 96338-5002

REPORT DOCUMENTATION PAGE

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED	
20230302	Final	START DATE 20200828	END DATE 20220827

4. TITLE AND SUBTITLE
Knowledge Externalization from Image-to-Image Translation

5a. CONTRACT NUMBER	5b. GRANT NUMBER	5c. PROGRAM ELEMENT NUMBER
	FA2386-20-1-4044	
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**

6. AUTHOR(S)
Jaegul Choo

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Korea Advanced Institute of Science and Technology
291 Daehak-ro, Yuseong-gu
Taejon
KR

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AOARD UNIT 45002 APO AP 96338-5002	AFRL/AFOSR IOA	AFRL-AFOSR-JP-TR-2023-0064

12. DISTRIBUTION/AVAILABILITY STATEMENT
A Distribution Unlimited: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
We present a novel Animation CelebHeads dataset (AnimeCeleb) to address an animation head reenactment. Different from previous animation head datasets, we utilize a 3D animation models as the controllable image samplers, which can provide a large amount
of head images with their corresponding detailed pose annotations. To facilitate a data creation process, we build a semi-automatic pipeline leveraging an open 3D computer graphics software with a developed annotation system. After training with the AnimeCeleb,
recent head reenactment models produce high-quality animation head reenactment results, which are not achievable with existing datasets. Furthermore, motivated by metaverse application, we propose a novel pose mapping method and architecture to
tackle a cross-domain head reenactment task. During inference, a user can easily transfer one's motion to an arbitrary animation head. Experiments demonstrate a usefulness of the AnimeCeleb to train animation head reenactment models, and the superiority of our crossdomain head reenactment model compared to state-of-the-art methods. Our dataset and code are available at https://github.com/kangyeolk/AnimeCeleb. This work has been published in ECCV'22.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES
a. REPORT	b. ABSTRACT	c. THIS PAGE		
U	U	U	SAR	15

19a. NAME OF RESPONSIBLE PERSON	19b. PHONE NUMBER *(Include area code)*
AKIRA NAMATAME	3152277010

Standard Form 298 (Rev.5/2020)
Prescribed by ANSI Std. Z39.18

Annual Report for AOARD Grant FA9550-21-S-0001

Knowledge Externalization from Image-to-Image Translation

March 2, 2023

Name of Principal Investigators: Assoc. Prof. Jaegul Choo
- e-mail address : jchoo@kaist.ac.kr
- Institution : Korea Advanced Institute of Science and Technology (KAIST)
- Mailing Address : 291 Daehak-ro, N24 LG Innovation Hall, Room# 3109, Yuseong-gu, Daejeon 34141, South Korea
- Phone : +82-42-350-1813
- Fax : +82-42-350-1813

Period of Performance: 05/17/2022-05/16/2023

Abstract:

We present a novel Animation CelebHeads dataset (AnimeCeleb) to address an animation head reenactment. Different from previous animation head datasets, we utilize a 3D animation models as the controllable image samplers, which can provide a large amount of head images with their corresponding detailed pose annotations. To facilitate a data creation process, we build a semi-automatic pipeline leveraging an open 3D computer graphics software with a developed annotation system. After training with the AnimeCeleb, recent head reenactment models produce high-quality animation head reenactment results, which are not achievable with existing datasets. Furthermore, motivated by metaverse application, we propose a novel pose mapping method and architecture to tackle a cross-domain head reenactment task. During inference, a user can easily transfer one's motion to an arbitrary animation head. Experiments demonstrate a usefulness of the AnimeCeleb to train animation head reenactment models, and the superiority of our cross-domain head reenactment model compared to state-of-the-art methods. Our dataset and code are available at https://github.com/kangyeolk/AnimeCeleb. This work has been published in ECCV'22.

Contents

1 Introduction

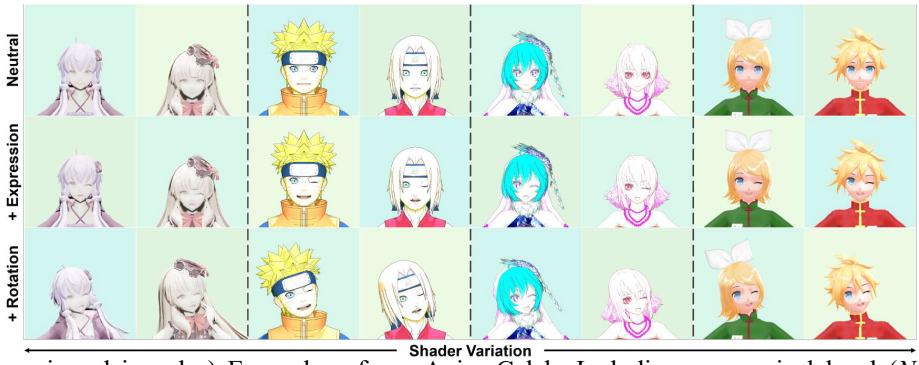


Fig. 1: (Better viewed in color) Examples of our AnimeCeleb. Including a canonical head (*Neutral*), the AnimeCeleb contains expression-changed (+*Expression*) and head-rotated (+*Rotation*) images with varying shaders.

Recent head reenactment methods [4,15,17] show impressive results on control- ling a human head motion after trained with large-scale human talking head video datasets [5,14]. The common approaches [21,20,17,15] for this task is to learn diverse motion changes between two contiguous frames, which require a large amount of head videos to train a high-performing neural network model. Due to the dependency of human video datasets, such approaches show weak generalization capacity on the animation domain, because animation characters have distinct appearances (*e.g.*, explicit lines and large eyes) compared to the human head ones. Our key contribution is to construct a large-scale animation head dataset, AnimeCeleb, for head reenactment, which deems as a data-centric solution to produce high-quality reenactment results on the animation domain.

Obviously, a standard approach to build an animation dataset would be to collect the images from comic books and cartoon films. Instead, we propose a principled manner to construct animation dataset, where 3D animation models serve as valuable image samplers. This leads to three following benefits. First, we can ceaselessly simulate the specified pose[5] of a 3D animation model, enabling to generate an *unlimited* number of multi-pose images of the same identity. Second, the simulated poses are easily obtainable as detailed pose vectors, where each dimension represents an individual semantic of an expression or a head angle. Lastly, a 3D vector graphics environment gives freedom to render the *arbitrary* resolution images with various shaders (See Fig. 1 horizontal axis). These strengths bring multiple use cases including the animation head reenactment and intuitive pose editing.

Technically, our data creation process involves 3D animation model collection, semantic annotation and image rendering. In this process, we first collect the 3D animation models spanning a wide range of animation characters. The collected 3D models contain a set of morphs that can deform appearances of the 3D models in face and body part. To identify suitable morphs relevant to the head reenactment task, we develop an annotation system to filter the expression-irrelevant morphs. We employ Blender[6] that can execute codes for a head detection and a pose manipulation to enable an automatic image rendering. A great interest of an animation domain is to transfer a user's motion to the animation character, which is potentially applicable in a metaverse and a virtual avatar system. In this paper, we focus on transferring a user's pose to the animation character, and refer to this problem as a *cross-domain head reenactment task*. A plausible solution to the task is building a shared pose representation space across the domains (*i.e.*, human and animation). We use 3D morphable model (3DMM) parameters as the shared pose representation, which is widely used in recent numerous head reenactment studies [7,22,15,18,8]. 3DMM is a parametric face modeling method that provides powerful tools for describing human heads with semantic parameters.

Since the AnimeCeleb pose vector is not compatible with 3DMM, we newly propose a *pose-mapping* method to transform an AnimeCeleb pose vector to 3DMM parameters. To be specific, we compute a set of distinct 3DMM parameters to describe the semantics that the AnimeCeleb includes, and

combine it to obtain 3DMM parameters corresponding to a AnimeCeleb pose vector. Owing to the pose mapping, we can guarantee that both the AnimeCeleb and VoxCeleb [14], a human head video dataset, share the pose representations. Furthermore, we propose a new architecture called an animation motion model (*AniMo*), in which datasets from different domains are used to learn how to manipulate a head image according to the motion residing in the shared representations. In this manner, our model is capable of transferring a human head motion represented as 3DMM parameters to an animation head.

In summary, our contributions to animation research are as follows:

- We propose a *novel data creation pipeline* and present a *public large-scale animation head dataset* AnimeCeleb, which contains groups of high-quality images and their corresponding pose vectors.
- We newly propose a *pose-mapping* method and a cross-domain head reenactment model *AniMo*, which jointly lead to a seamless motion transfer from a **human head** to an **animations head**.
- We demonstrate the effectiveness of AnimeCeleb in training head reenactment baselines, and experimental results show the superiority of *AniMo* on cross-domain head reenactment compared to state-of-the-art methods.

2 Animation CelebHeads Dataset

We first describe each step of the data creation of the AnimeCeleb in Section 2.1. Next, AnimeCeleb properties and statistics are given in Section 2.2. In Section 2.3, we show the animation head reenactment results on the AnimeCeleb and other animation datasets.

2.1 Data Creation Process

Fig. 2 depicts the overall process of the data creation pipeline. In the following, we provide details of each step from (A) to (D).

Data Collection (A). We collected 3D animation models from two different web sites: DevianArt (https://www.deviantart.com/) and Niconi solid(https://3d.nicovideo.jp/). Since all 3D animation models are copyrighted by their creators, we carefully confirmed the scope of rights and obtained permission from reachable authors. Finally, we acquired 3613 usable 3D animation models in total. We will release all 3D animation model *artists' list* along with the AnimeCeleb to acknowledge the credits of the artists. The collected 3D animation models contain two essential components. The first component is the **morphs** that can alter appearances of a 3D animation model on face or body parts. We are able to change an individual morph's continuous value ranging from [0, 1], and obtain a transformed appearance of a 3D animation model; for example, an animation head with open mouth in 0.3 proportion can be generated. The second one is the **bones** that can control head angles (*i.e.*, yaw, pitch and roll axes). In specific, the head angles are controlled by applying a rotation matrix to the neck bone.

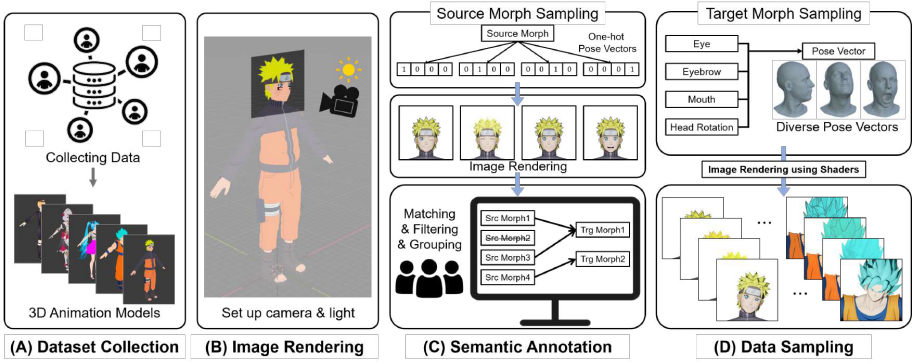
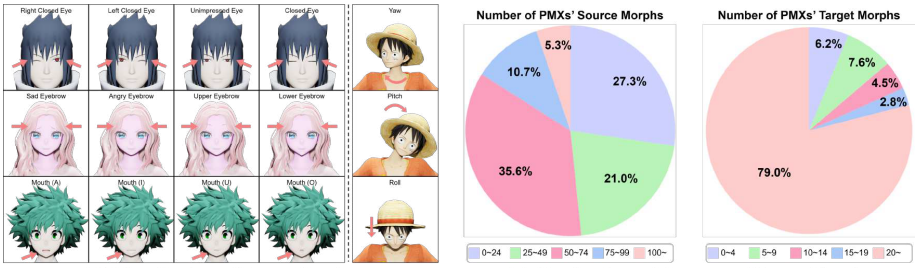


Fig. 2: **Dataset Creation Pipeline Overview.** 3D animation models are collected from two different websites (A). Then, a head part of the collected model is rendered after applying a morph with maximum intensity (B); these are then used for semantic annotation (C). In a data sampling step, sampled target morphs are used to compose pose vectors that serve as conditions to produce multi-pose images with diverse facial expressions and head rotations.

Image Rendering (B). To achieve an automatic sampling using 3D animation models, we develop a 2D head image creation pipeline built on Blender: an open-source 3D computer graphics software that supports the visualization, manipulation and rendering of 3D animation models. To successfully render the animation head images in Blender, we need to consider three aspects: (1) camera position, (2) light condition, and (3) image resolution. We set the camera position based on a neck bone position with the aim of capturing the head part. In respect to the light condition, we use a directional light point along the negative **y**-axis: frontal direction of an animation character (See Fig. 2 (B)). Before rendering, we set the resolution of the images as 256 × 256, which is a standard resolution used in previous head reenactment methods [17,15]. Nonetheless, since the AnimeCeleb images are rendered from a 3D vector graphics model, we can create a higher image resolution (*e.g.*, 1024 × 1024). To demonstrate its extensive usage, we present various generated samples under different conditions in the supplementary material. Note that the rendered images contain an alpha channel as a transparent background, which can separate the foreground animation character and the background.

Semantic Annotation (C). Each 3D animation model has a significantly different number of morphs ranging from zero to even over 100. However, a morph naming convention is different according to a creator, which makes it difficult to apply a standardized criterion before annotating an accurate semantic of an individual morph. A goal of the semantic annotation is to *identify* expression-related morphs and *annotate* the morphs according to the unified naming convention. Importantly, this allows to sample a properly functioning expression-related source morph from a 3D animation model during rendering. For example, when a morph あ attached to a specific 3D animation model is identified as indicating a semantic of pronouncing the syllable 'ah' with a mouth, then it can be annotated as the target morph (*i.e.*, Mouth (A)). After annotation, that source morph あ of the 3D model is used, when the target morph Mouth (A) is determined to control the mouth shape.



(A) Visualization of Target Morphs (B) Distributions of Source and Target Morphs

Fig. 3: (A) Visualizing target morphs' examples and head rotation. (B) The percentage of the number of source and target morphs on 3D animation models. The number of source morphs are widely distributed ranging from 0 to over 100, and most animation models have dense usable annotations (*i.e.*, target morphs).

To achieve the semantic annotation, we first define 23 *target* morphs, these are deemed as meaningful semantics to represent the facial expressions. We select the target morphs out of candidates collaborated with animation experts who work with cartoon makers. Fig. 3 (A) shows the examples of the target morphs that include meaningful semantics for three parts: eyes, eyebrows, and a mouth. Conversely to the target morphs, we denote the original morphs as *source* morphs in the remainder of this section. Next, we attempt to match the source morphs to the target morphs. Fortunately, a group of the source morphs with the identical name tends to portray the same semantics. Therefore, we take a two-stage approach: a group annotation and an individual inspection. The former collectively match a group of the source morphs under the same name to a target morph; the latter is responsible for inspecting the matched source morphs one-by-one to confirm whether it works correctly. During the group annotation, we count the number of source morphs that 3D models have, and remove the source morphs under 50. The individual inspection reduces the erroneous annotations that occur at the group annotation.

For this, we first render the head images after applying the entire source morphs independently and a neutral image without applying any morph using a 3D animation model (Fig. 2 (C) upper part). Afterwards, we match a group of source morphs to one of the target morphs (*i.e.*, group annotation) and correct the results in a single morph-level via comparing a neutral and a morph-applied image for each source morph (*i.e.*, individual inspection). The entire procedure is conducted on the newly developed annotation system (Fig. 2 (C) lower part).

Data Sampling (D). Throughout the data sampling, randomly selected target morphs for each part (*i.e.,* eyes, eyebrows and a mouth) are applied to a 3D animation model. The magnitudes of the morphs are determined by sampling from a uniform distribution, $U(0, 1)$, independently. In respect to the head rotation, a 3D rotation matrix is computed taking yaw, pitch and roll values sampled between -20° and 20°. We render a transformed head after applying the morphs and the rotation, and acquire a paired pose vector $\mathbf{p} \in R^{20}$. A real-time rendering engine that Blender provides is used to produce the manipulated images and paired pose vectors. During rendering, we utilize 4 different types of shaders as shown in Fig. 2 to provide diverse textured 2D images. Since the morphs and the head rotation are applied independently, two image groups: a group of frontalized images with expression (*frontalized-expression*) and head rotated images with expression (*rotated-expression*) are included in the AnimeCeleb. The number of images sampled from the 3D model are determined differently depending on the number of annotated target morphs that a 3D animation model has. When a 3D animation model contains more than five annotated target morphs, we generate 100 images; if not (*e.g.*, zero), just 20 images are obtained.

Dataset	Num. of Images	Identity Labels	Face Align.	Unified Style	Image Source	Attribute Anno.
Kaggle Anime Face [1]	63K	✗	✗	✗	Media	-
Danbooru 2019 [3]	302K	✓	✗	✗	Media	-
iCartoonFace [23]	0.39M	✓	✗	✗	Media	3D Head Pose, Bounding Box Gender
AnimeCeleb (Ours)	2.4M	✓	✓	✓	3D Models	3D Head Pose, Expression, Foreground Mask Artistic Style

Table 1: Comparison between the AnimeCeleb and public animation head datasets.

Fig. 4: (A) Head reenactment results trained with the iCartoonFace that bear an identity leakage problem. (B) An intra-variation within the same identity of the iCartoonFace is extremely large. (C) Average inception score comparison on three datasets; the average scores using 1000 identities indicate that iCartoonFace contains relatively inconsistent styles within the identity than those of the VoxCeleb and the AnimeCeleb.

2.2 Dataset Description

AnimeCeleb Properties. Fig. 3 (A) shows the examples of multiple target morphs for each part and head rotation results. The target morphs consist of 9 eye-related morphs, 9 eyebrow-related morphs and 5 mouth-related morphs. Note that the pre-defined target morphs include the semantics related to both eyes or eyebrows, which fill two values (*e.g.*, left and right eye) of a 17-dimensional pose vector (*expression* part). In total, 3613 different 3D models are used to generate the AnimeCeleb. As can be seen in Fig. 3 (B) left, the number of source morphs of collected raw 3D animation models are widely

distributed, averaging 49 morphs. After the semantic annotation, most animation models have more than 20 target morphs as shown in Fig. 3 (B) right; this indicates the source morphs are densely matched to the target morphs.

Comparison with Other Datasets. As shown in Table 1, the AnimeCeleb has three advantages compared to the public existing animation head datasets [1,3,23]. The advantages mainly stem from exploiting the power of 3D software and 3D animation models. First, detailed annotations such as facial expressions and head rotations can be easily gained because we are able to manipulate the head using our morph annotation (Table 1 Attribute Anno.). Second, the AnimeCeleb provides a massive amount of animation images that have unified styles (Table 1 Num. of Images, Unified Style). We believe that these properties help to develop high-performing neural networks in broad applications. Lastly, the AnimeCeleb contains four different unified styles in consideration of different cartoon textures. A similar approach [13] has been proposed using 3D animation models to construct an animation face dataset and achieve promising results on head reenactment. The contribution of AnimeCeleb is the first publicly available dataset that contains animation faces with pose annotations as well as the data sampling pipeline.

2.3 Animation Head Reenactment

Overview. The head reenactment aims to transfer a pose from a driving image to a source image. A common training scheme of the head reenactment model is to extract a pose from a driving image, and feed it with a source image to a decoder to reconstruct the driving image. Therefore, training a high-performing head reenactment model requires a large-scale video dataset, containing a set of the same identity images that can serve as a source and driving image pair. In a human domain, the VoxCeleb [14], a large-scale talking head dataset, plays this role. We believe that the AnimeCeleb is analogous to the VoxCeleb in an animation domain, which bears a potential to train a high-performing animation head reenactment model.

Prior head reenactment approaches are categorized into two groups whether a pre-computed pose annotation is utilized during training or not. The FOMM does not use the pose annotation, and learn relative motion between two images to convey the pose to a source image. In contrast, numerous studies [21,15,20] take advantage of the pose annotations such as keypoints and 3DMM parameters obtained from off-the-shelf pose extractors. Among them, we train two representative head reenactment baselines [17,15] from each category with the AnimeCeleb: the FOMM [17] and the PIRenderer [15], which uses 3DMM parameters to describe a head pose.

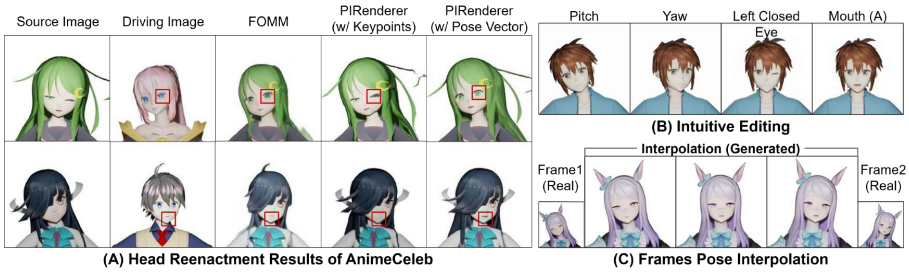


Fig. 5: (A) Qualitative results of the FOMM and PIRenderer trained with the AnimeCeleb. (B) Intuitive editing of an animation head image with different pose vectors. (C) Filling in-between frames using linearly interpolated pose vectors.

Model	Same-Identity		Cross-Identity
	FID↓	SSIM↑	FID↓
FOMM	23.45	0.824	29.94
PIRenderer(w/ keypoints)	27.84	0.770	21.48
PIRenderer(w/ pose vector)	**20.27**	**0.826**	**16.52**

Table 2: Quantitative results of animation head reenactment. Obviously, for the AnimeCeleb dataset, the PIRenderer trained with pose vector outperforms the PIRenderer with keypoints and the FOMM.

Experiment Setup. When training the PIRenderer, we replace 3DMM with the pose vectors of the AnimeCeleb. For the dataset comparison, we additionally train the baselines [17,15] using the iCartoonFace[23]. Although there exist other animation head datasets [1,3], we select the iCartoonFace as a comparison dataset, acknowledging the size of it and accurate identity labels. Furthermore, with the aim of pose annotation comparison, we train the PIRenderer leveraging the keypoints for both datasets. We utilize an off-the-shelf animation keypoint detector[10] that gives 28 keypoints of an animation head image. All implementations are conducted following the hyperparameters denoted the papers with 3319 train set and 294 test dataset created with the first shader style.

We evaluate the trained models on (1) Self-identity task where the same character provides the source and driving image, and (2) Cross-identity task where two frames of different character sampled from the AnimeCeleb serve as the source and driving image. For evaluation, Fréchet Inception Distance (FID) [11] and Structural Similarity (SSIM) [19] are adopted to measure the generated images quality. Note that the AnimeCeleb is applicable to other existing head reenactment models [20,9,21] that need image keypoints, yet we implement two representative baselines here.

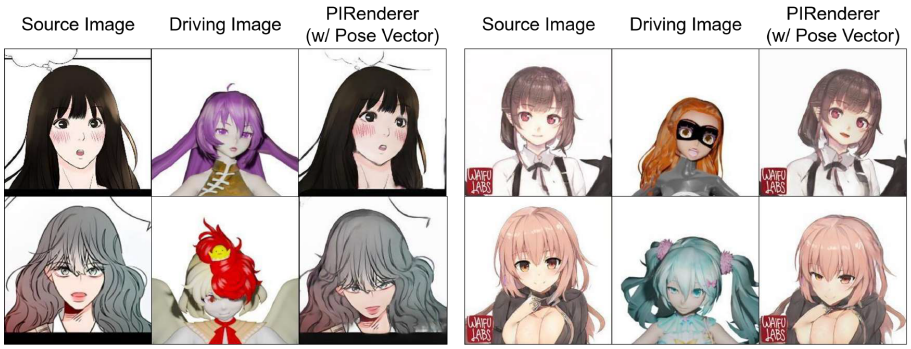


Fig. 6: Head reenactment results on other animation datasets.

Experimental Results with the iCartoonFace. Fig. 4 (A) shows the cross-identity head reenactment outputs of two models trained with the iCartoonFace. Despite the attempts to train the FOMM and PIRenderer with the iCartoonFace, we have found that the trained models show poor performance, producing blurry outputs. We assume that excessive variation within a single identity is the main cause of the results. In fact, considering that the iCartoonFace consists of the images collected from different appearance scenes, most images have own properties as seen in Fig. 4 (B). For quantitative analysis, we measure the Inception Score (IS) [16] by averaging 1,000 image sets of the same identity. As seen in Fig. 4 (C), we confirm that the iCartoonFace records higher IS score, compared to the VoxCeleb and the AnimeCeleb. This indicates that the iCartoonFace contains unacceptable appearance complexity, hence learning from such images goes beyond the capacity of existing head reenactment models.

Advantage of Pose Annotation. As seen in Fig. 5 (A), the FOMM trained with the AnimeCeleb produces plausible outputs, yet still has undesirable deformation. Different from it, the trained PIRenderers successfully preserve the source head structure while imitating a given driving image

with both pose annotations (*i.e.*, keypoints and pose vector). Especially, the PIRenderer(w/ pose vector) accurately conveys a driving pose to the source image as shown in Fig. 5 (A) red boxes. It is because the AnimeCeleb pose vectors hold more direct guidance (*e.g.*, 80% mouth openness) than the keypoints. This results can be quantitatively confirmed in Table 2, where the PIRenderer(w/ pose vector) outperforms other baselines on both same-identity and cross-identity head reenactment tasks. Besides, the PIRenderer(w/ pose vector) is able to intuitively edit head poses based on given pose vectors (Fig. 5 (B)) and generate the in-between frames by interpolating the pose vectors of two different frames (Fig. 5 (C)).

Other Animation Results. We demonstrate the generalization capacity of the trained model on other animation datasets. In an experiment, we evaluate the PIRenderer(w/ pose vector) on different collected head datasets including WaifuLabs(https://waifulabs.com/) and Naver Webtoons(https://comic.naver.com/). As seen in Fig. 6, the model successfully transfer a given driving image pose to an animation head.

3 Cross-Domain Head Reenactment

Overview. Although we show a promising animation head reenactment result in Section 2.3, controlling characters' head pose as a human user wants (*i.e.*, cross-domain head reenactment) is another important application that bears a potential to be used in a virtual YouTuber system and a cartoon production. In this section, we address the cross-domain head reenactment using the proposed pose mapping method and the *AniMo*.

In a standard head reenactment training scheme, two frames are sampled from a video: a source image s and driving image d, and reconstruct d. Different from previous methods [17,15], we leverage two videos from different domains, respectively. Since a direct supervision across domains is not available during training, the source and driving image pair from animation domain: $s^{(a)}$, $d^{(a)}$ and human domain: $s^{(r)}$, $d^{(r)}$ are utilized to reconstruct the driving images, $d^{(a)}$ and $d^{(r)}$, respectively. In the following, we illustrate the details of a driving pose representation (Section 3.1). Then, we describe a training pipeline and its objective functions (Section 3.2).

Difference from PIRenderer. Our architecture design is inspired by PIRenderer [15], yet two novel components, a pose-mapping method, and separate domain-specific networks, are proposed to improve cross-domain head reenactment performance. The pose-mapping method enables to align blendshape and 3DMM, which gives the capability to handle a pose from human domain (*i.e.*, cross domain). Also, the domain-specific networks help to preserve a given source image's textures for each domain, and improve the quality of image. Note that our pose-mapping method can help PIRenderer to improve the performance on cross-domain head reenactment task.

3.1 Driving Pose Representations

Human Pose Representation. Our approach employs the 3DMM parameters to describe a pose of a driving human head image. With the 3DMM, a 3D human face shape $\mathbf{S}$ can be represented as $\mathbf{S} = \bar{\mathbf{S}} + \alpha\mathbf{B}_{id} + \beta\mathbf{B}_{exp}$, where $\bar{\mathbf{S}}$ is the average face shape, $\mathbf{B}_{id}$ and $\mathbf{B}_{exp}$ denote the principal components of identity and expression based on 200 scans of human faces [2], respectively. Also, $\alpha \in \mathbb{R}^{80}$ and $\beta \in \mathbb{R}^{64}$ indicate the coefficients that control the relative magnitude between the facial shape and expression basis. The head rotation and translation are defined as $\mathbf{R} \in SO(3)$ and $\mathbf{t} \in \mathbb{R}^3$. We use a pre-trained 3D face reconstruction model [6] to extract the 3DMM parameters from the human head images. Discarding α for excluding an identity-related information, we only exploit a subset space of the 3DMM parameters M to represent a human head pose, where $\mathbf{m} \in M$ comprises of expression coefficients, head rotation and translation: $\mathbf{m} \equiv \{\beta, \mathbf{R}, \mathbf{t}\} \in \mathbb{R}^{70}$.

Pose Mapping. The AnimeCeleb pose vector $\mathbf{p} \in \mathbb{R}^{20}$ consists of independent coefficients $\mathbf{b} \in B$ and head angles $\mathbf{h} \in H$, where B denotes a 17-dimensional space of concatenated expression coefficient and H indicates a 3D head angle space. In this step, we aim at discovering a mapping relationship from the AnimeCeleb pose vector to the 3DMM parameters. To this end, we propose a pose mapping function: $T : B \times H \rightarrow M$, which is responsible to find its corresponding 3DMM parameters, given a

pose vector. We construct a direct mapping relationship between the coefficients $\mathbf{b}$ and the 3DMM expression parameters β using facial landmarks as a proxy space and expressing each coefficient's semantics via manually manipulating the landmark positions.

In the following, we elaborate the details step-by-step with Fig. 7 (A). (T.0) Before the landmark manipulation, we first obtain an initial landmark position, which corresponds to a neutral 3DMM coefficient. To be specific, the initial landmark position is obtained from a rendered mesh with setting the entire 3DMM coefficients as $\mathbf{o}$ expressed as $\{\alpha_o, \beta_o, \mathbf{R}_o, \mathbf{t}_o\}$, meaning that the average face shape $\bar{\mathbf{S}}$ at center location offers the initial landmark position. (T.1) Next, the initial landmarks are manipulated according to each semantic; for example, *left closed eye* landmarks can be achieved by minimizing the distances between the upper and the lower eyelid keypoints at the left eye. (T.2 and T.3) Then, the manipulated landmarks l^k with k-th semantic are used to update the initial β by minimizing the ℓ_2 distance between l^k and the landmarks extracted from the rendered mesh using β. Also, we employ a ℓ_2 regularization during updating β. Completing this process for each landmark, we can gain the fitted 3DMM expression parameters for each semantic: $\Phi = \{\beta^k\}_{17} \in \mathbb{R}^{17\times 64}$. Finally, the pose mapping function can be written as: $\mathbf{m}_i = T(\mathbf{b}_i, \mathbf{h}_i) = (\mathbf{b}_i \cdot \Phi) \oplus \Pi(\mathbf{h}_i) \oplus \mathbf{o} \in M$, where Π denotes a mapping to convert a degree into radian measurement and $\oplus, i$ indicate a concatenation operation and a data index, respectively. In addition, $\mathbf{o} \in \mathbb{R}^3$ is concatenated to represent translation parameters.

3.2 Training Pipeline

Fig. 7 depicts an overview of our framework, which consists of three networks described below.

Motion Network. Given a driving pose $\mathbf{m}$, our motion network F generates a latent pose code $\mathbf{z} \in Z$, where Z denotes a latent pose space. Formally, this can be written as: $\mathbf{z}^{(a)} = F(\mathbf{m}^{(a)})$, $\mathbf{z}^{(r)} = F(\mathbf{m}^{(r)})$, where $\mathbf{m}^{(a)} = T(\mathbf{b}, \mathbf{h})$ is a transformed driving pose corresponding to the driving image $d^{(a)}$ in an animation domain and $\mathbf{m}^{(r)}$ denotes a subset of 3DMM parameters obtained from the driving image $d^{(r)}$ in a human domain, respectively. Thanks to the pose mapping method, the motion network F can be designed as *domain-agnostic* manner. The learned latent pose code $\mathbf{z}$ is transformed to estimate the affine parameters for adaptive instance normalization (AdaIN) [12] operations. The pose information parameterized as the affine parameters plays a role in predicting an optical flow in the warping network W and injecting a fine-detailed pose in the editing network G.

Warping & Editing Network. For sake of simplicity, we omit the domain notation unless needed, such as $\mathbf{z} = \{\mathbf{z}^{(a)}, \mathbf{z}^{(r)}\}$, $d = \{d^{(a)}, d^{(r)}\}$, and $s = \{s^{(a)}, s^{(r)}\}$ in the descriptions of warping and editing network. Inspired by the PIRenderer [15], we employ *domain-specific* warping networks and an editing network for each domain. A warping network W takes a source image s and latent pose code $\mathbf{z}$ to predict the optical flow $\mathbf{u}$ that approximates the coordinate offsets to reposition a source head alike a driving head.

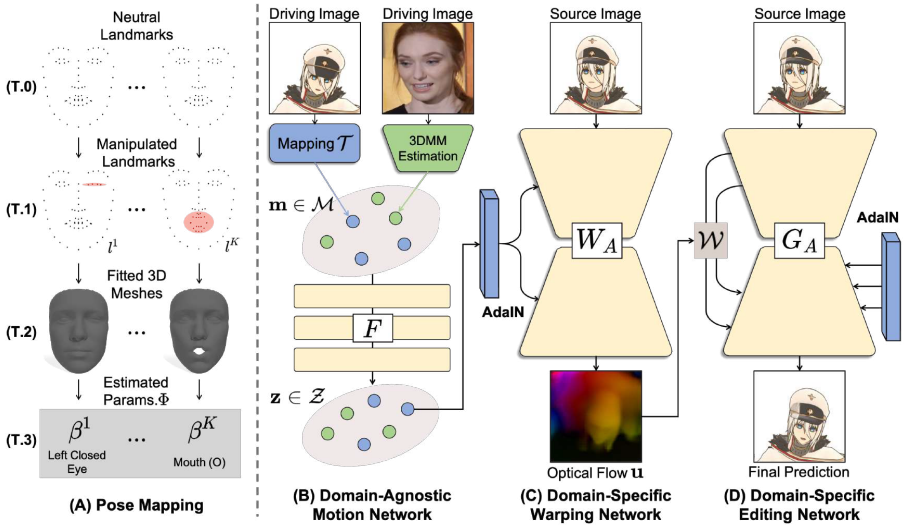


Fig. 7: Overview of (A) pose mapping method and (B)-(D) *AniMo*. information parameterized as the affine parameters plays a role in predicting an optical flow in the warping network *W* and injecting a fine-detailed pose in the editing network *G*.

Next, the source image is fed into an encoder part of an editing network *G* and the optical flow $\mathbf{u}$ is applied to the intermediate multi-scale feature maps. This leads to spatial deformation of the feature maps according to the driving pose. During decoding in *G*, the AdaIN operation is used to inject the pose informa- tion. After training, the warping network mainly focuses on causing a large pose, including the head rotation, whereas the editing network serves to portrait a detailed expression-related pose. We train our framework with a reconstruction loss and a style loss following the PIRenderer [15]. The architecture, implementation details and objective functions are elaborated in the supplementary material.

3.3 Experiments

Experiment Setup. Different from Section 2.3, we use both cartoon texture shader style AnimeCeleb and the VoxCeleb[14] as a training dataset. The VoxCeleb contains 22,496 talking-head videos collected from online videos, and we use downloadable 18,503 videos for the train set and 504 videos for test set. We evaluate the trained models on self-identity, and cross-domain head reenactment where the images of the AnimeCeleb and the VoxCeleb alternatively serve as a source and a driving image respectively. Similar to Section 2.3, FID and SSIM are used to assess the quality of generated images. In addition, we introduce a Head Angle Error (HAE) that measures the ℓ_1 distances between the driving image's head angles and those of the generated image with the aim of evaluating head rotating ability. To be specific, we take advantage of a pre-trained head angle regressor, based on ResNet-18 [10] architecture and trained with the AnimeCeleb train set using ℓ_1 distance objective function between a predicted angle and the ground-truth $\mathbf{h}$. In experiments, we use randomly sampled 1,000 pairs of source and driving images to compute evaluation metrics.

Train Dataset	Methods	Self-identity (AnimeCeleb)		Self-identity (VoxCeleb)		Cross-domain (Vox.→Anime.)		Cross-domain (Anime.→Vox.)
		FID↓	SSIM↑	FID↓	SSIM↑	FID↓	HAE↓	FID↓
Single Dataset (VoxCeleb)	FOMM	47.91	0.648	**16.10**	**0.803**	122.83	0.177	94.23
	PIRenderer	134.91	0.532	19.67	0.604	95.75	0.176	96.42
	LPD	-	-	-	-	166.54	0.171	-
Joint Datasets (AnimeCeleb, VoxCeleb)	FOMM	45.01	**0.748**	19.60	0.748	144.88	0.196	126.49
	PIRenderer+$\mathcal{T}$	16.07	0.735	18.98	0.611	69.80	0.195	61.67
	Ours	**16.05**	0.738	19.34	0.606	**18.78**	**0.128**	**41.04**

Table 3: Quantitative comparison with baselines on self-identity and cross-domain head reenactment tasks. The expression A → B denotes that transferring an A's motion to B's a source image.

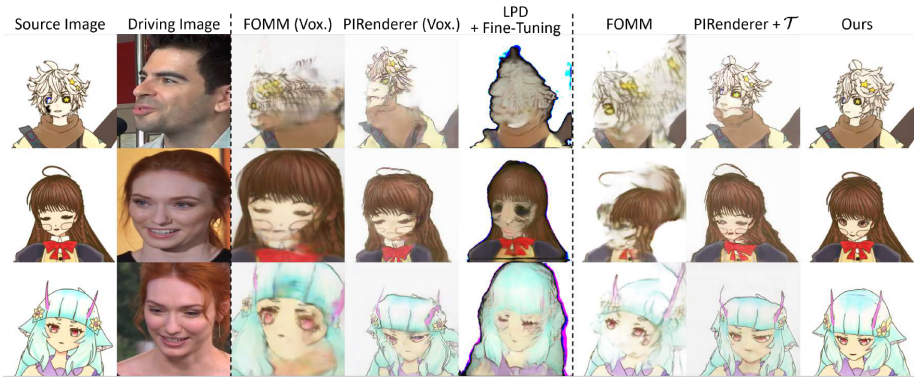


Fig. 8: Qualitative comparison between our model and the baselines.

Comparison with State of the Art. We compare the the *AniMo* with state-of-the-art models [4,17,15] quantitatively and qualitatively. Since we leverage two datasets during training, comparable baselines are trained on either the VoxCeleb following their original implementations or both the VoxCeleb and AnimeCeleb. During evaluation, we make an inference of manipulated animation or human head by optionally leveraging the decoder of each domain.

Table 3 shows quantitative comparisons between the AniMo and the baselines on the self-identity and the cross-domain head reenactment. When evaluating the self-identity head reenactment within the AnimeCeleb, it is obvious that the models trained on both the AnimeCeleb and the VoxCeleb surpass those trained on the VoxCeleb. On the contrary, quantitative results on self-identity head reenactment within the VoxCeleb demonstrate that joint datasets may be harmful to the reconstruction task. Unlike these results, our model outperforms all baselines on cross-domain head reenactment tasks in terms of an image quality and an imitating head pose, indicating the superiority of our model in transferring a pose across the domains. Fig. 8 shows qualitative comparisons between the *AniMo* and the baselines on the cross-domain head reenactment. The FOMM, which relies on the unsupervised landmarks, does not work well, because the model attempts to align the appearance of the source image as the driving image's head structure, and this leads to the identity leakage problem as well as introducing blurring artifacts. In contrast, the PIRenderer and latent pose descriptor (LPD) [4], where the pose is injected by the AdaIN operations, successfully retain a head structure of the source image, yet produce rather blurry outputs. As seen in the PIRenderer+$\mathcal{T}$, the blurry artifacts can be improved by incorporating the AnimeCeleb as an additional training dataset with the pose mapping $\mathcal{T}$. Meanwhile, our model clearly outperforms the baselines, preserving more vivid textures of the source image and accurately reflecting the pose of the driving image with the aid of the domain-specific networks. We conclude that the shared pose space introduced by the pose mapping and the domain-specific design help the model to transfer the pose across domains.

4 Conclusions

4.1. Summary and future work

In this paper, we present the AnimeCeleb, a large-scale animation head dataset, which is a valuable and practical resource for developing animation head reenactment model. Departing from existing animation datasets, we utilize 3D animation models to construct our animation head dataset by simulating facial expressions and head rotation, resulting in neatly organized animation head dataset with rich annotations. For this purpose, we built a semi-automatic data creation pipeline based on Blender and a semantics annotation tool. We believe that the AnimeCeleb would boost and contribute to animation-related research. On the other hand, we propose the pose mapping and architecture to address cross-domain head reenactment to admit transferring a given human head motion to an animation head. Conducted experiments demonstrate the effectiveness of the *AniMo* on cross-domain head reenactment and intuitive image editing. In the future work, we plan to extend the AnimeCeleb and develop more advanced cross-domain head reenactment model.

4.2. Publication outcomes (attached as a separate pdf file)

AnimeCeleb: Large-Scale Animation CelebHeads Dataset for Head Reenactment:
Kangyeol Kim,* Sunghyun Park,* Jaeseong Lee,* Sunghyo Chung, Junsoo Lee, and **Jaegul Choo** (*: equal contributions) European Conference on Computer Vision (**ECCV**), 2022, *Accepted* (28.4% acceptance rate).

References

1. Kaggle animation face. https://www.kaggle.com/splcher/animefacedataset
2. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3d faces. In: Proceedings of the 26th annual conference on Computer graphics and interactive techniques. pp. 187–194 (1999)
3. Branwen, G., Anonymous, Community, D.: Danbooru2019: A large-scale anime character illustration dataset. https://www.gwern.net/Crops (May 2020)
4. Burkov, E., Pasechnik, I., Grigorev, A., Lempitsky, V.: Neural head reenactment with latent pose descriptors. In: Proc. of the IEEE conference on computer vision and pattern recognition (CVPR). pp. 13786–13795 (2020)
5. Chung, J.S., Nagrani, A., Zisserman, A.: Voxceleb2: Deep speaker recognition. arXiv preprint arXiv:1806.05622 (2018)
6. Deng, Y., Yang, J., Xu, S., Chen, D., Jia, Y., Tong, X.: Accurate 3d face re- construction with weakly-supervised learning: From single image to image set. In: Proc. of the IEEE conference on computer vision and pattern recognition workshop (CVPRW) (2019)
7. Gafni, G., Thies, J., Zollhofer, M., Nießner, M.: Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In: Proc. of the IEEE conference on computer vision and pattern recognition (CVPR). pp. 8649–8658 (2021)
8. Guo, Y., Chen, K., Liang, S., Liu, Y.J., Bao, H., Zhang, J.: Ad-nerf: Audio driven neural radiance fields for talking head synthesis. In: Proc. of the IEEE international conference on computer vision (ICCV). pp. 5784–5794 (2021)
9. Ha, S., Kersner, M., Kim, B., Seo, S., Kim, D.: Marionette: Few-shot face reen- actment preserving identity of unseen targets. In: Proc. the AAAI Conference on Artificial Intelligence (AAAI). vol. 34, pp. 10893–10900 (2020)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. of the IEEE conference on computer vision and pattern recognition (CVPR). pp. 770–778 (2016)
11. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. In: Proc. the Advances in Neural Information Processing Systems (NeurIPS) (2017)
12. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE international conference on computer vision. pp. 1501–1510 (2017)
13. Khungurn, P.: Talking head anime from a single image 2: More expressive. http://pkhungurn.github.io/talking-head-anime-2/ (2021)
14. Nagrani, A., Chung, J.S., Zisserman, A.: Voxceleb: a large-scale speaker identification dataset. arXiv preprint arXiv:1706.08612 (2017)
15. Ren, Y., Li, G., Chen, Y., Li, T.H., Liu, S.: Pirenderer: Controllable portrait im- age generation via semantic neural rendering. In: Proc. of the IEEE international conference on computer vision (ICCV). pp. 13759–13768 (2021)
16. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. Proc. the Advances in neural information processing systems (NeurIPS) **29** (2016)
17. Siarohin, A., Lathuili`ere, S., Tulyakov, S., Ricci, E., Sebe, N.: First order motion model for image animation. Proc. the Advances in Neural Information Processing Systems (NeurIPS) **32**, 7137–7147 (2019)

18. Wang, C., Chai, M., He, M., Chen, D., Liao, J.: Cross-domain and disentangled face manipulation with 3d guidance. arXiv preprint arXiv:2104.11228 (2021)
19. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
20. Zakharov, E., Ivakhnenko, A., Shysheya, A., Lempitsky, V.: Fast bi-layer neural synthesis of one-shot realistic head avatars. In: Proc. of the European Conference on Computer Vision (ECCV). pp. 524–540. (2020)
21. Zakharov, E., Shysheya, A., Burkov, E., Lempitsky, V.: Few-shot adversarial learn- ing of realistic neural talking head models. In: Proc. of the IEEE conference on computer vision and pattern recognition (CVPR). pp. 9459–9468 (2019)
22. Zhang, C., Zhao, Y., Huang, Y., Zeng, M., Ni, S., Budagavi, M., Guo, X.: Facial: Synthesizing dynamic talking face with implicit attribute learning. In: Proc. of the IEEE international conference on computer vision (ICCV). pp. 3867–3876 (2021)
23. Zheng, Y., Zhao, Y., Ren, M., Yan, H., Lu, X., Liu, J., Li, J.: Cartoon face recognition: A benchmark dataset. In: Proceedings of the 28th ACM International Conference on Multimedia. pp. 2264–2272 (2020)