EXPRESSION DOMAIN TRANSLATION NETWORK FOR CROSS-DOMAIN HEAD REENACTMENT

Taewoong Kang[1,*] *Jungsik Oh*[2,*] *Jaeseong Lee*[2] *Sunghyun Park*[2] *Jaegul Choo*[2]

[1] Korea University [2] KAIST

ABSTRACT

Despite the remarkable advancements in head reenactment, the existing methods face challenges in cross-domain head reenactment, which aims to transfer human motions to domains outside the human, including cartoon characters. It is still difficult to extract motion from out-of-domain images due to the distinct appearances, such as large eyes. Recently, previous work introduced a large-scale anime dataset called AnimeCeleb and a cross-domain head reenactment model including an optimization-based mapping function to translate the human domain's expressions to the anime domain. However, we found that the mapping function, which relies on a subset of expressions, imposes limitations on the mapping of various expressions. To solve this challenge, we introduce a novel *expression domain translation network* that transforms human expressions into anime expressions. Specifically, to maintain the geometric consistency of expressions between the input and output of the expression domain translation network, we employ a *3D geometric-aware loss function* that reduces the distances between the vertices in the 3D mesh of the input and output. By doing so, it forces high-fidelity and one-to-one mapping with respect to two cross-expression domains. Our method outperforms existing methods in both qualitative and quantitative analysis, marking a significant advancement in the field of cross-domain head reenactment.

***Index Terms*—** Cross-domain, Head Reenactment

1. INTRODUCTION

Given the advancements in online live streaming platforms such as YouTube, there has been a growing trend among users to express themselves through virtual avatars (*i.e.*, virtuber). This trend has elevated the significance of head reenactment tasks, wherein human motions are transferred to other virtual characters, in response to such user needs. Recent studies [1, 2, 3, 4, 5] in head reenactment have made it possible to transfer human motions onto other human heads, by leveraging large-scale datasets of human talking head videos. However, existing head reenactment methods still face challenges when applied to virtual avatars, such as anime characters, which exist outside the human domain.

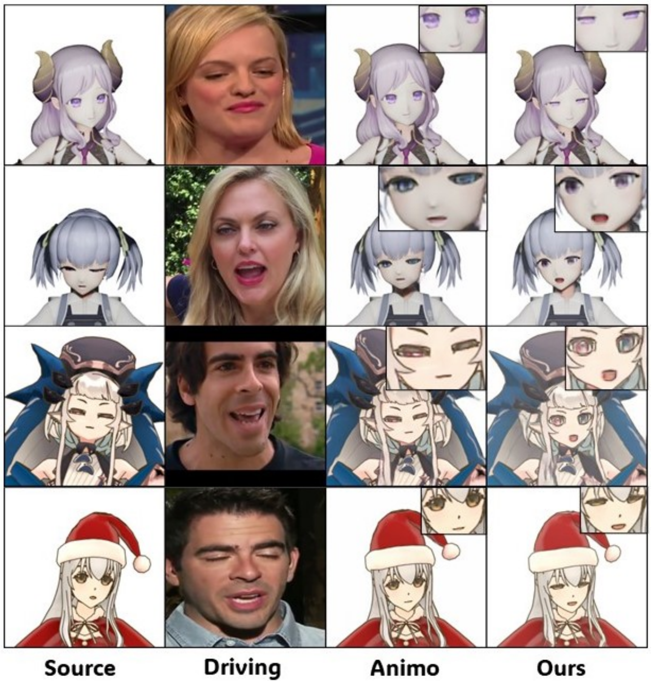

Fig. 1. Cross-domain face reenactment examples of our method. Given the anime image, we edit it by injecting the pose and expression from driving image.

Such a process of transferring human head motion to images from domains outside the human is referred to as cross-domain head reenactment. Conventional head reenactment approaches typically encode the human motion through landmarks [1, 2], 3D morphable models (3DMM) [3], or latent descriptors [6]. However, extracting motion from out-of-domain images (*e.g.*, cartoon characters) with distinct appearances, including small mouths or large eyes, is challenging. Moreover, due to the shortage of suitable video datasets for head reenactment for other domains, it is difficult to facilitate cross-domain head reenactment.

To address these challenges, only a few works [7, 8, 9, 10] have attempted to tackle cross-domain head reenactment. Recently, AnimeCeleb [7] has endeavored to facilitate head reenactment of cartoon characters by constructing a large-scale animation dataset based on 3D character models, including pairs of 2D cartoon images and pose vectors. Furthermore, they proposed a cross-domain head reenactment approach leveraging two domain datasets (*i.e.*, AnimeCeleb and VoxCeleb [11]). Specifically, AnimeCeleb has designed an optimization-based mapping function to transform AnimeCeleb's pose vectors into the 3DMM space, leveraging landmarks of specific expressions (*e.g.*, left closed eye). Even when trained solely on AnimeCeleb, the model is applicable to a wide range of cartoon images with various styles, such as Waifu Labs [1], Naver Webtoon [2] and 2D Disney [3]. However, we discovered that during the process of mapping expressions of AnimeCeleb's pose vector to the 3DMM spaces, the optimization-based method relying on only a subset of expression imposes limitations on mapping various expressions, as illustrated in Fig. 1.

To overcome the lack of reflecting facial expressions caused by the limitations, we propose a novel cross-domain expression translation network to map the human expressions to the anime expressions. This network is designed to transform 3DMM parameters into semantically equivalent pose vectors. The main idea revolves around the utilization of a *3D geometric-aware loss*. This loss function operates as a proxy within a shared vertex space, which constitutes a 3D mesh, facilitating the learning process. Consequently, we achieve the one-to-one mapping from 3DMM parameters to semantically equivalent pose vectors. This way, our approach not only retains the merits of AnimeCeleb but also excels in the accurate transmission of facial expressions. Through experiments, we demonstrate the superiority of our method in the field of cross-domain talking heads.

2. METHODOLOGY

In this section, we present our *cross-domain head reenactment framework*. Given the driving image $\mathbf{I}_d$ of the human domain with the 3DMM vector $\mathbf{p}$, our model aims to generate anime image $\hat{\mathbf{I}}$ by modifying the head pose and face expressions of the source image $\mathbf{I}_s$ of the anime domain.

2.1. Expression Space Domain Gap

We employ a DECA [12] to extract FLAME [13] parameters to encode human motion. Specifically, with FLAME parameters, human face mesh can be represented as:

$$\mathbf{T}_P(\beta,\theta,\psi) = \mathbf{T} + B_S(\beta;S) + B_P(\theta;P) + B_E(\psi;E), \quad (1)$$

where the average mesh shape $\mathbf{T}$ in zero pose, $\mathbf{B}_S(\beta;S)$, $B_P(\theta;P)$, and $B_E(\psi;E)$ denote shape, head angle, and expression blendshapes. However, we only utilize expression coefficients $\psi \in \mathbb{R}^{50}$ and head pose $\theta \in \mathbb{R}^6$, where $\theta = [pose; jaw]; pose \in \mathbb{R}^3; jaw \in \mathbb{R}^3$.

Basically, we leverage the AnimeCeleb dataset [7], which consists of pairs of anime images and pose vectors. While for encoding anime motion, we utilize AnimeCeleb's pose vectors $\mathbf{v} \in \mathbb{R}^{20}$, which consist of 17-dimensional expression coefficients $\mathbf{b} \in \mathcal{B}$ and head angles $\mathbf{h} \in \mathcal{H}$, following the previous work [7]. Specifically, expression coefficients $\mathbf{b}$ consist of six eye-related dimensions, six eyebrow-related dimensions, and five mouth-related dimensions. For instance, the first dimension of the pose vector $\mathbf{v}$ corresponds to a left-eye wink, and thus it holds a value within the range of 0 to 1, varying according to the degree of the eye being closed.

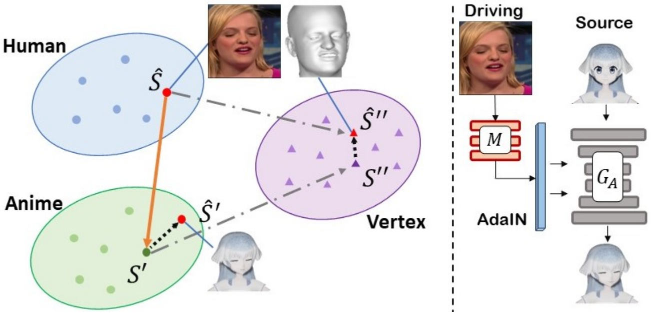

Fig. 2. (left) Diagram of our expression domain translation network where $\hat{S}$ for ground truth and S for prediction. (right) Overview of our framework

There exist differences in the representation of facial pose and expression between the human and anime. Therefore, for effectively transferring human motion to anime images, it is crucial to accurately map FLAME parameters encoding human motion to pose vectors representing anime motion. Notably, while head angles allow for precise one-to-one mapping as described in the previous work [7], the approach for translating human's expression coefficients $\psi \in \mathbb{R}^{50}$ into anime's expression coefficients $\mathbf{b} \in \mathbb{R}^{17}$ is required.

2.2. Expression Domain Translation Network

To map the human's expression coefficients ψ into anime's expression coefficients $\mathbf{b}$, we propose *expression domain translation network* $\mathbf{M}$. Specifically, with ψ and jaw as input variables that are related to the expressions, the model outputs the expression coefficients $\mathbf{b}$. Given that FLAME incorporates expressions in conjunction with the jaw to construct the mesh, the use of jaw becomes an indispensable component. When we train the network, a paired dataset that denotes equivalent facial expressions on two different domains is required. However, due to the cross-domain issue, it becomes imperative to train the model using an unpaired dataset. To address this issue, we utilize a *pose adapter* and train the expression domain translation network $\mathbf{M}$ with a *3D geometric aware loss*.

Pose Adapter. To map the anime's expression coefficients $\mathbf{b}$ onto the vertex space, it needs to be converted into ψ form. Consequently, we employed the use of a pose adapter $\mathcal{T}$. In order to obtain $\mathcal{A}$, we first identified landmarks corresponding to each dimension of the pose vector. Subsequently, we performed optimization to determine the ψ values that contain these landmarks. Then, we can get the fitted ψ for each semantic: $\varphi = \{\psi^k\}_{k=1}^{17} \in \mathbb{R}^{17\times 50}$. Finally, the pose adapter

[1] https://waifulabs.com/

[2] https://comic.naver.com/

[3] https://toonify.photos/

$\mathcal{A}$ can be written as: $\mathbf{p}_i = \mathcal{A}(\mathbf{b}_i) = \mathbf{b}_i \cdot \varphi \in \mathbf{P}$ containing geometrically meaningful information. The difference between the mapping function $\mathcal{T}$ from Animo [7] and the pose adapter $\mathcal{A}$ is that $\mathcal{T}$ maps to 3DMM [14] expression parameters $\beta \in \mathbb{R}^{64}$, while our pose adapter maps to FLAME [13] expression parameters $\psi \in \mathbb{R}^{50}$. Unlike $\mathcal{T}$, which uses the mapping directly, we only used our pose adapter to send it to the vertex space due to distribution mismatch.

3D Geometric Aware Loss. To identify the anime's expression coefficients $\mathbf{b}$ that is semantically equivalent to ψ, we train the expression translation network $\mathbf{M}$ utilizing 3D geometric aware loss. Due to the expression space domain gap, we map both $\mathbf{b}$ and ψ to the vertex space, which is mutually compatible space. Additionally, by utilizing vertex space, we are able to employ information that is geometrically aware. Exploiting these characteristics, we train the parameters from each domain to possess expressions that are geometrically congruent.

First, we employ vertex loss to ensure that the corresponding vertices have identical coordinates. Formally, the loss is given as

$$\mathcal{L}_{ver} = \frac{1}{n}\sum_{i=1}^{n} ||\hat{\mathbf{t}}_i - \mathbf{t}_i||^2, \tag{2}$$

where $\mathbf{t}_i \in \mathbb{R}^3$ is vertices that consists mesh. Additionally, we extract just the 68 keypoints $\mathbf{k}_i$ from vertices to further train on important information.

$$\mathcal{L}_{lm} = \sum_{i=1}^{68} \parallel \hat{\mathbf{k}}_i - \mathbf{k}_i \parallel_1 . \tag{3}$$

To better capture sensitive and important features like the eyes and mouth, we have applied an eye and mouth closure loss. The eye closure loss computes the relative offset of landmarks $\mathbf{k}_i$ and $\mathbf{k}_j$ on the upper and lower eyelid, and measures the difference to the offset of the corresponding predicted landmark $\mathbf{p}_i$ and $\mathbf{p}_j$. Similarly, the mouth closure loss computes the upper and lower outer mouth's offset distance. The loss is defined as

$$\mathcal{L}_{eye} = \sum_{(i,j)\in E} \parallel |\hat{\mathbf{k}}_i - \hat{\mathbf{k}}_j| - |\mathbf{k}_i - \mathbf{k}_j| \parallel_1, \tag{4}$$

$$\mathcal{L}_{mouth} = \sum_{(i,j)\in M} \parallel |\hat{\mathbf{k}}_i - \hat{\mathbf{k}}_j| - |\mathbf{k}_i - \mathbf{k}_j| \parallel_1, \tag{5}$$

where E is the set of upper and lower eyelid landmark pairs and M is the set of upper/lower outer mouth landmark pairs. In summary, our full objective function is given as:

$$\mathcal{L}_{total} = \mathcal{L}_{lm} + \mathcal{L}_{eye} + \mathcal{L}_{mouth} + \lambda_{ver} \cdot \mathcal{L}_{ver}. \tag{6}$$

Here, λ_{ver} is the hyperparameter and set to 100.

2.3. Anime Generator

Fig. 2 provides an overview of our framework. In this section, we introduce the remaining part of our framework, which synthesizes anime images based on the pose vectors predicted from the expression domain translation network.

Motion Network. With a driving pose $\mathbf{v}$, the motion network F generates a latent pose code $z \in \mathcal{Z}$, where $\mathcal{Z}$ denotes a latent pose space. Thanks to our network $\mathbf{M}$ and the characteristics of $\mathbf{v}$, the motion network F can be designed as the domain-agnostic and controllable method, which is the main difference with PIRender. Then, we just need the generator that can edit the source image with the given z.

Warping & Editing Network. With warping and editing network, we can generate image that is guided by z through adaptive instance normalization (AdaIN) [15]. A warping network predicts the optical flow $\mathbf{u}$ that serves to approximate the coordinate offsets to reposition a source head like a driving head. An editing network that serves to portrait a detailed expression-related pose gets source image, optical flow $\mathbf{u}$, and latent pose code z. See PIRenderer [3] for details.

With our expression domain translation network and the anime generator, we are capable of achieving state-of-the-art performance in cross-domain head reenactment.

3. EXPERIMENTS

3.1. Experiment Setup

Datasets. To train our expression domain translation network $\mathbf{M}$, we take a subset of videos from Voxceleb [11]. We downloaded 18,503 videos for train set and 504 videos for test set. Also, we use AnimeCeleb [7] dataset to train anime generator.

Training Details. The expression domain translation network $\mathbf{M}$ and anime generator $\mathbf{G}$ are trained separately. For the expression domain translation network $\mathbf{M}$, we trained it for 50 epochs, where the batch size is 512 and the optimizer is Adam with an learning rate of 1×10^{-4}. For the generator, we trained the model for 200 epochs, where the batch size is 8 and the optimizer is Adam with an learning rate of 1×10^{-4}.

3.2. Comparison with Baselines

We compare our model with state-of-the-art models quantitatively and qualitatively. Moreover, we have empirically substantiated the efficacy of our model with respect to its distributional characteristics.

Quantitative Evaluation. Table 1 shows quantitative comparisons between our model and the baselines [1, 3, 7] on the cross-domain face reenactment. When evaluating cross-domain face reenactment, we found that existing metrics do not adequately capture facial expressions. Therefore, we introduce a new metric, called the Keypoint Distance Ratio (KDR), which measures the ℓ_1 distance ratio of eyes compared with neutral keypoints' distance. The reason why we compare the relative distance of eyes is that there is topological heterogeneity between human and anime character domain (*e.g.,* Anime character's abstract distance of eye's lid is innately larger than humans'). Specifically, this ratio compares the upper and lower eyelid distances in both the driving

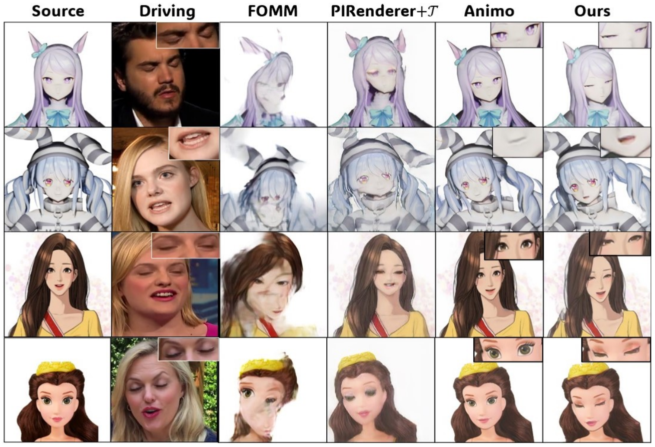

Fig. 3. Qualitative comparison between our model and the previous method Animo [7] on cross-domain face reenactment given the source image from AnimeCeleb [7], Naver Webtoon, 2D Disney and the driving image from VoxCeleb [11]

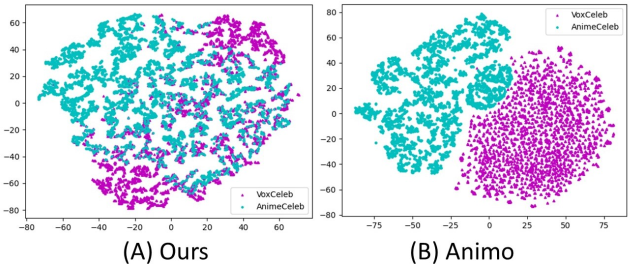

Fig. 4. Distribution concurrency check through t-SNE. (A) Ours shows a lower expression distribution discrepancy than (B) Animo.

image and the predicted image. For a more detailed analysis, we employ a keypoint detector designed for the anime domain [16]. Because the keypoints in anime are different from those in humans, we measure the ℓ_1 distance and compare ratios to bridge this domain gap. Despite being trained solely on a single dataset, thereby benefiting from a reduced model size and diminished training time, our model excels by achieving superior scores in both frechet inception distance (FID) and KDR metrics.

Qualitative Evaluation. Fig. 3 shows qualitative comparisons between our model and the baselines. In the realms of texture and identity preservation, it is discernible that both our method and Animo [7] clearly outperform FOMM [1] and PIRenderer [3]. Moreover, while Animo exhibits limited capability in faithfully capturing facial expressions, our method demonstrates a markedly superior performance in accurately reflecting them. We conclude that the expression domain translation network **M** serves as an instrumental component, facilitating the successful cross-domain transfer of facial expressions within the model. More qualitative results and video results available on the project website [4]

Train Dataset	**Model**	**Cross-Domain**	
		FID$_{\downarrow}$	KDR$_{\downarrow}$
Joint Dataset (Vox, AnimeCeleb)	FOMM	100.95	N/A
	PIRenderer + $\mathcal{T}$	49.55	N/A
	Animo	28.69	0.466
Single Datasets (AnimeCeleb)	Ours	**23.15**	**0.236**

Table 1. Quantitative results of animation face reenactment. For FOMM [1] and PIRenderer [3], not available for KDR because of distorted output making keypoint detection impossible.

Loss			
Vertex	Landmark	E&M Dist.	**KDR**$_{\downarrow}$
✓			0.2431
	✓		0.2394
✓	✓		0.2390
✓	✓	✓	**0.2360**

Table 2. Ablation study on the loss component. E&M Dist. indicates eye and mouth closure loss.

Distribution Concurrency. One of the key factors contributing to the efficacy of our method is the successful alignment of distributions across both domains. To empirically validate this, we conducted a t-SNE analysis on a sample size of 5,000 data points within the input space of the motion network. The results can be visualized in Fig.4.

3.3. Ablation Study

To empirically substantiate the necessity of our loss function, we conducted a comprehensive ablation study by dropping the loss component. As evident from Table 2, the model that was subjected to the full complement of loss terms demonstrated the highest performance in KDR. Moreover, even the configuration that was trained solely with the vertex loss outperformed Animo [7], thereby substantiating the efficacy of imposing loss constraints within the 3D spatial domain.

4. CONCLUSION

In this paper, we propose a novel cross-domain expression translation network to map the human expressions to the anime expressions. We achieve a significant improvement in the performance of cross-domain neural talking heads by implementing a shared 3D vertex space as a learning proxy. Our model's superiority is validated through both quantitative and qualitative evaluations. As a direction for future work, we aim to train the network on a dataset of human expressions, allowing the mapping network to mapping function as an explicit semantic controller.

[4] https://keh0t0.github.io/research/Animetalkinghead/

5. REFERENCES

[1] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe, "First order motion model for image animation," *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, pp. 7137–7147, 2019.

[2] Ting-Chun Wang, Arun Mallya, and Ming-Yu Liu, "One-shot free-view neural talking-head synthesis for video conferencing," in *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2021, pp. 10039–10049.

[3] Yurui Ren, Ge Li, Yuanqi Chen, Thomas H Li, and Shan Liu, "Pirenderer: Controllable portrait image generation via semantic neural rendering," in *Proc. of the IEEE international conference on computer vision (ICCV)*, 2021, pp. 13759–13768.

[4] Nikita Drobyshev, Jenya Chelishev, Taras Khakhulin, Aleksei Ivakhnenko, Victor Lempitsky, and Egor Zakharov, "Megaportraits: One-shot megapixel neural head avatars," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 2663–2671.

[5] Bowen Zhang, Chenyang Qi, Pan Zhang, Bo Zhang, HsiangTao Wu, Dong Chen, Qifeng Chen, Yong Wang, and Fang Wen, "Metaportrait: Identity-preserving talking head generation with fast personalized adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 22096–22105.

[6] Egor Burkov, Igor Pasechnik, Artur Grigorev, and Victor Lempitsky, "Neural head reenactment with latent pose descriptors," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 13786–13795.

[7] Kangyeol Kim, Sunghyun Park, Jaeseong Lee, Sunghyo Chung, Junsoo Lee, and Jaegul Choo, "Animeceleb: Large-scale animation celebheads dataset for head reenactment," in *Proc. of the European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 414–430.

[8] Aayush Bansal, Shugao Ma, Deva Ramanan, and Yaser Sheikh, "Recycle-gan: Unsupervised video retargeting," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 119–135.

[9] Linsen Song, Wayne Wu, Chaoyou Fu, Chen Qian, Chen Change Loy, and Ran He, "Everything's talkin': Pareidolia face reenactment," *arXiv preprint arXiv:2104.03061*, 2021.

[10] Borun Xu, Biao Wang, Jinhong Deng, Jiale Tao, Tiezheng Ge, Yuning Jiang, Wen Li, and Lixin Duan, "Motion and appearance adaptation for cross-domain motion transfer," in *European Conference on Computer Vision*. Springer, 2022, pp. 529–545.

[11] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[12] Yao Feng, Haiwen Feng, Michael J Black, and Timo Bolkart, "Learning an animatable detailed 3d face model from in-the-wild images," *ACM Transactions on Graphics (ToG)*, vol. 40, no. 4, pp. 1–13, 2021.

[13] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero, "Learning a model of facial shape and expression from 4d scans.," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 194–1, 2017.

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

[15] Xun Huang and Serge Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1501–1510.

[16] hysts, "Anime face detector," `https://github.com/hysts/anime-face-detector`, 2021.