Synthetic to Authentic: Transferring Realism to 3D Face Renderings for Boosting Face Recognition

Parsa Rahimi[1,2], Behrooz Razeghi[2], and Sébastien Marcel[2,3]

[1] École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
[2] Idiap Research Institute, Martigny, Switzerland
[3] Université de Lausanne (UNIL), Lausanne, Switzerland
parsa.rahiminoshanahg@epfl.ch,{behrooz.razeghi,marcel}@idiap.ch

Fig. 1: 3D-Rendered images of human faces [6] (left image in each column), and post-processing images by image-to-image translation (right image in each column) for boosting the performance of a Face Recognition trained on the synthetic data.

Abstract. In this paper, we investigate the potential of image-to-image translation (I2I) techniques for transferring realism to 3D-rendered facial images in the context of Face Recognition (FR) systems. The primary motivation for using 3D-rendered facial images lies in their ability to circumvent the challenges associated with collecting large real face datasets for training FR systems. These images are generated entirely by 3D rendering engines, facilitating the generation of synthetic identities. However, it has been observed that FR systems trained on such synthetic datasets underperform when compared to those trained on real datasets, on various FR benchmarks. In this work, we demonstrate that by transferring the realism to 3D-rendered images (i.e., making the 3D-rendered images look more real), we can boost the performance of FR systems trained on these more photorealistic images. This improvement is evident when these systems are evaluated against FR benchmarks like IJB-C, LFW which utilize real-world data by 2% to %5, thereby paving new pathways for employing synthetic data in real-world applications. The project page is available at: https://idiap.ch/paper/syn2auth.

Keywords: Image-to-Image Translation · Face Recognition Systems · Realism Transfer · 3D-Rendered Datasets · Photorealism in Synthetic Data

1 Introduction

Given the increasing dependency on artificial intelligence (AI) systems in our everyday lives, it becomes essential to comprehend and rectify any potential problems that might arise within these systems. A primary issue with today's systems is their strong dependency on large volumes of data required for training. This dependency presents numerous problems, both ethically and legally, in areas such as vision and language models. For instance, datasets often collected from web crawls may contain ethically and legally sensitive content, with inherently uncontrollable and inaccurate labels. This issue becomes even more critical in the sensitive task of Face Recognition (FR) systems, which requires the collection of personal and sensitive image modalities containing faces. Furthermore, considering legal policies such as GDPR and other digital ethics guidelines [1, 23], the use of existing datasets like WebFace260M [55] and CASIA-WebFace [51] could be problematic when deployed in critical applications. Besides these concerns, the necessity to collect large sample sizes for training an effective deep FR model poses another challenge. Therefore it is crucial to address these issues, which involve one of the most important applications of AI systems in our daily lives: FR systems [11,21] (e.g., unlocking our phones, security gates). Due to the mentioned problems with data captured from the real world, there has been an increase in research exploring the applicability of synthetic data as an alternative or complement to real datasets in various computer vision problems [6, 14, 24]. For instance, studies using a 3D rendering pipeline [46] have shown that for tasks like Face Parsing and Landmark Localization, the accurate labels provided by rendering pipelines can surpass the performance of models trained on real datasets in landmark localization tasks (since the images are rendered using a model-based face, the landmark locations are accurate compared to those in human-annotated datasets collected from the real world).

Recently, authors in [5] have demonstrated that by using the conditional generation of different classes with a pre-trained denoising diffusion model [4, 39], it is possible to boost the performance of downstream classification tasks, emphasizing the potential benefits of using synthetic data to enhance AI models.

As mentioned earlier, collecting large datasets for specific computer vision tasks can be challenging, especially in the domain of facial images, which are considered one of the most sensitive data modalities. To alleviate this problem, there has been a surge in research within the community focused on developing methodologies for creating datasets that either complement existing ones [30] (mainly for bias mitigation, addressing the problem of underrepresented data for some sensitive groups) or entirely replace the datasets used for training FR systems. Methods such as IDiffFace [8], Digiface1M [6], and DCFace [22] aim to generate useful datasets for training an FR system from scratch. To generate a useful dataset for training an FR system, we need to include various identities with diverse demographic labels (i.e., inter-class variability on the order of tens of thousands), and for each identity, variations of the same identity (i.e., intra-class variability, such as different poses and expressions, etc.). When generating variations of the same identity, it is crucial to ensure the preservation of the

identity, for example, when changing the pose. Current methods in literature enforce this condition by using a separate, strong, pre-trained FR system [8,22] or by utilizing identity attribute labels in large datasets like CASIA-WebFace [51]. However, the challenge lies in *replacing* the training dataset of the FR system with synthetic data, not using a strong pre-trained FR system trained on real data to generate synthetic data which is a strong and unreasonable prior.

It is difficult to quantify the benefits gained through synthetic datasets, as they often fall short of the performance achieved by the pre-trained FR systems used during their generation phase.

Another approach involves using 3D-rendering engines, as seen in publicly available datasets like Digiface1M [6]. This method is advantageous because it does not require any specific enforcement for identity preservation when generating variations of the same identity, given direct access to the exact mesh and vertices that will eventually be rendered into a face image using different rendering methodologies.

Hence, we can conclude that by changing the pose of the subject or the lighting of the environment, the identity remains unchanged. However, a significant downside is observed when training an FR system with these 3D-rendered datasets, like Digiface1M, and evaluating it against standard FR benchmarks such as IJB-C [26]. There exists a large performance gap, possibly due to an Out-of-Distribution (OOD) problem [6].

1.1 Research Problem

The collection of datasets containing identity-labeled human faces is often impeded by privacy concerns [32]. Consequently, there is an increasing trend toward synthesizing such data, which is then utilized to train FR models. This paper investigates the following *hypothesis*:

> Face images in existing rendered datasets can be made more realistic while preserving identities, **without** the need for **identity labels** or a **pre-trained FR model**, thereby improving the accuracy of FR models trained on this data.

Our primary contribution is to validate this hypothesis through extensive experiments. It tries to address the OOD problem of 3D Face Renderings compared to face images captured from the real world.

1.2 Key Contributions

In this paper, our key contribution lies in investigating and analyzing the potential of introducing photorealism into 3D-rendered datasets, as depicted in Figure 2, *without using any identity labels or a trained FR system*. We demonstrate that we can achieve a performance gain with an FR system trained on our more photorealistic dataset (i.e., transferring realism), thereby opening new avenues for exploring this topic. To the best of our knowledge, this is the first

attempt to study the effect of photorealism on top of 3D-rendered facial images for gaining performance improvement in FR systems.

Our contributions are as follows:

- We analyze the applicability of transfer learning methodologies to bridge the gap between imperfect simulation of the real world in the 3D rendering engines, specifically in the domain of face images.
- In contrast to previous works, which require a strong pre-trained FR model to generate useful data for training an FR model, we observe a performance boost without relying on any pre-trained FR system or identity labels in the challenging task of FR.
- We introduce a mathematical formulation for the realism transfer idea and reformulate other approaches using this unified framework.

In Section 2, we lay some background on the problem and introduce relevant methods to our analysis. In Section 3, we define our problem setting. Finally, in Section 4, we explain our experimental analysis.

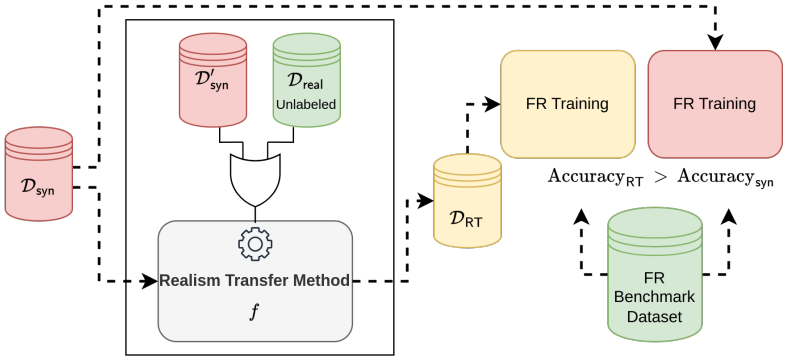

Fig. 2: In this paper, we study the efficacy of image-to-image translation methodologies applied to enhance the performance of face recognition—in essence a challenging classification task. Starting with a dataset of 3D-rendered human faces (*i.e.* $\mathcal{D}_{\text{syn}}$), that exhibit a domain shift compared to real-world human face images, we apply various image-to-image translation and Face Restoration methodologies (*i.e.*, **Realism Transfer Method** Block) that only require limited **identity unlabeled** real datasets (*i.e.*, $\mathcal{D}_{\text{real}}$) **or** subset of unrealistic images (*i.e.*, $\mathcal{D}'_{\text{syn}}$) themselves to train. We then train a face recognition network on both the original (unrealistic-looking) and the newly translated (more realistic) images, $\mathcal{D}_{\text{RT}}$, to investigate whether this approach can improve the accuracy of FR systems.

2 Related Work

In this section, we will briefly overview relevant topics related to the usage and generation of synthetic data, as well as methods that can be applied to our problem setting.

2.1 Synthetic Data in Computer Vision

Synthetic data generation has become a key strategy for creating vast quantities of accurately annotated data, which is necessary for computer vision tasks that often require detailed labeling. This approach facilitates the development of comprehensive datasets essential for training and improving vision-based algorithms and has been extensively explored in the research community recently. For example, synthetic data has been utilized in tasks such as Semantic Image Segmentation [7,24,44], Optical Flow Estimation [41], Face Parsing [46] and Face Recognition [6], Human Motion Understanding [15, 25], and other computer vision tasks that require dense, accurate labels. Some of these approaches [6,14,46] utilize 3D-rendering engines and physics simulators [10] to model the underlying physics of the real world. This ensures that the distribution of the generated data is similar to that of data gathered from the real world, making it useful for the applicability of these data and the models trained on them. Our analysis in this paper makes a significant stride in alleviating the domain gap caused by imperfect simulation, modeling, and the limited computing power available to simulate the real world.

2.2 Unpaired Image-to-Image Translation

In this section, we briefly highlight methodologies that are particularly promising for enhancing realism in computer graphics applications—a critical challenge in the domain of FR. Among these, VSAIT [42] introduces a novel method for unpaired image-to-image translation using Vector Symbolic Architectures (VSA) to minimize semantic flipping, which occurs when the content of the translated images does not match the semantic context of source domain. This is specifically important as it plays a key role in the photorealism of computer graphics applications [34]. The authors propose leveraging the VSA framework's capacity for high-dimensional symbolic computation to maintain content consistency between the source and translated images. This is especially useful since the VSA framework is robust against noise. This method is one of the methods that we examine for the *Realism Transfer Method* in Figure 2. In the Density Changing Regularized Unpaired Image Translation (DECENT) method [49], the authors focused on the concept of density-changing regularization. The method assumes that image patches of high probability density in one domain should be mapped to patches of high density in another domain. To enforce this principle, two density estimators were trained for each domain, and penalties were applied to the variance in density changes. This approach allows for more accurate preservation of neighboring information without relying on pairwise distances. Recently, authors in [20] introduced the Unpaired Neural Schrödinger Bridge (UNSB) method, which formulates the Schrödinger Bridge problem for the I2I task as a sequence of adversarial learning tasks. By leveraging discriminators and regularization techniques, they effectively overcome the curse of dimensionality. Essentially, their approach minimizes transport costs under constraints of Kullback-Leibler divergence.

2.3 Inverse Problem and Generative Prior

Among the approaches that incorporate a generative prior, inverse problem methodologies can also be applied to enhance realism. We consider two main types of generators: GAN-based and Diffusion-based. Specifically, in the case of employing StyleGANs [17, 19], which are trained on the domain of real data (*e.g.*, FFHQ [18] or its recent extension LPFF [47]), and inverting unrealistic images to one of the StyleGANs's latent-spaces (*e.g.*, $\mathcal{W}$, $\mathcal{W}^+$) using various methodologies, we aim to achieve the desired realism by reconstructing the resulting latent point. There are various methods for StyleGAN inversion, including *Optimization-Based* [2, 3], *Encoder-Based*, such as e4e [43] and pSp [33] and *HyperNetwork-Based* [13]. We leave the interested reader to recent surveys for details of each approach [48].

Diffusion models [4, 40] have recently emerged as a powerful new approach to generative modeling. In the diffusion process, these models introduce small amounts of noise to the original image in steps. During the reverse process, they attempt to estimate and remove the noise added to the original image. By repeating this process in the forward phase, we can transition from a signal domain to white Gaussian noise. In the denoising reverse process, it is possible to reconstruct the original signal. In the context of diffusion models, DDIM inversion [40] is a fundamental technique that introduces small increments of noise to a given image to approximate the corresponding input noise. Running a reverse diffusion with DDIM and this noise allows for the reproduction of the original image. In our problem setting, similar to StyleGAN Inversion, we utilize an unconditional diffusion model trained exclusively on a dataset like FFHQ. Our objective is to invert synthetic images back to a noise map and then reconstruct the input image. This approach allows us to uniquely bridge the distribution gap between real and synthetic images.

2.4 Face Restoration Methodologies

Face Restoration in computer vision aims to enhance degraded facial images through methods like super-resolution, denoising, and deblurring. Deep learning models, especially Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs), have shown significant advancements in addressing this problem. The authors in CodeFormer [54] applied the idea of vector quantization [45] to pre-train a quantized autoencoder through self-reconstruction, thereby obtaining a high-quality discrete codebook of face images and the corresponding decoder. The combination of the codebook's prior knowledge and the decoder is then used for face restoration. Based on this codebook prior, a Transformer is employed for the accurate prediction of code combinations from low-quality inputs. Additionally, a controllable feature transformation module is introduced to enable a flexible trade-off between the restoration quality and fidelity of the downgraded face images. The authors in PGDiff [50] introduced the concept of partial guidance, in which the diffusion prior acts as a regularization, and guidance is provided only on the desired properties of high-quality images. The key to [50] is constructing proper guidance for each task of restoration, inpainting, and masking separately. Methods like PGDiff [50] cannot be

directly applied to our problem setting, as they require the use of a pre-trained FR system for their restoration guidance.

2.5 Synthetic Data Generation for Face Recognition

The authors in SYNFace [29] utilize DiscoFaceGAN [12] to create facial images with detailed control over specific attributes such as identity, pose, expression, and illumination. This addresses the issue of limited variation within synthetic datasets, which impacts the performance of FR systems. By blending features from two synthetic identities to create new ones, SYNFace suggests a method to closely mimic real-world data, recommending a combination of synthetic and real images. In [9], the authors invert a dataset containing binary attribute labels of faces into the $\mathcal{W}$ space of a StyleGAN2 generator. They then fit a Support Vector Machine, using the distance to the hyperplane as a measure of the variation's scale. By moving in the direction perpendicular to the hyperplane for each attribute, they generated a small dataset to evaluate an FR system. As mentioned earlier, DigiFace-1M [6] provides a large-scale synthetic dataset for FR, produced through computer graphics. It uniquely defines each identity with specific facial details, allowing for varied expressions and environments. This model, which is independent of real data, narrows the gap between synthetic and real data, setting a new benchmark for accuracy. However, it faces challenges such as unrealistic textures and an unexamined demographic distribution.

DCFace [22], a newer Diffusion model, is designed for synthetic FR and features a two-stage process: generating synthetic identities and mixing these identities with styles from a "style bank." This approach demonstrates a strong capacity for creating unique and diverse identities, as evidenced by its performance in comparison with other approaches. However, as previously mentioned, the use of pre-trained FR systems or datasets with large identity is an unreasonable prior as the goal is to generate synthetic data for training FR system primarily. In IDiffFace [8], the authors introduced a method for generating synthetic datasets for face recognition by leveraging conditional Latent Diffusion Models (LDM) [35]. Significant emphasis is placed on the diffusion model's conditioning mechanism on face embeddings from a pre-trained FR system. This approach enables the creation of highly realistic and varied synthetic faces by conditioning the generative process on compact, identity-specific embeddings, albeit at the cost of utilizing a separate pre-trained FR system and the identity labels provided by large FR datasets. GANDiffFace [27] relies on the popular pre-trained model provided by Stable Diffusion. This approach comprises two steps: the first is dedicated to the synthesis of identities based on StyleGAN3 [17] and transformation in its latent space. This transformation is based on directions in the latent space that change specific attributes of images to introduce small intra-class variability, such as altering the pose. Subsequently, relying on the pre-trained text-to-image generator Stable Diffusion and the DreamBooth [36] personalization fine-tuning approach, they introduce more intra-class variability. The problem with this approach is its high reliance on large datasets [37] used to train Stable Diffusion, which are not privacy-friendly.

3 Transferring Realism to 3D Rendered Faces

3.1 Problem Formulation

Consider a dataset $\mathcal{D}_{\mathsf{syn}}$ comprising 3D-rendered images $\{\{\mathbf{x}_n^k\}_{k=1}^{K_n}\}_{n=1}^{N} \subseteq \mathcal{X}$ of human faces, consisting of N identities. For each identity, $n \in \{1, \cdots, N\}$, there exists an identity-dependent number of variations, K_n, representing different variations of the same identity. Let $P_{\mathbf{X}}$ denote the empirical probability distribution of the synthetic 3D-rendered data.

Our objective is to improve the utility of the synthetic dataset $\mathcal{D}_{\mathsf{syn}}$ with respect to a utility measure, by utilizing either an unlabeled real dataset $\mathcal{D}_{\mathsf{real}}$ with few samples, or a subset of the synthetic dataset $\mathcal{D}_{\mathsf{syn}}$ itself, denotes as $\mathcal{D}_{\mathsf{syn}}^{'}$, for training FR systems. In the following, we explore various approaches to post-process the synthetic dataset $\mathcal{D}_{\mathsf{syn}}$ to obtain a new dataset $\mathcal{D}_{\mathsf{RT}}$ for training the FR systems. We denote the generating distribution of post-processed data by $P_{\mathbf{Y}}$.

Consider two measurable spaces $\mathcal{X}$ and $\mathcal{Y}$, where $\mathcal{X}$ represents the domain of 3D-rendered images (source), and $\mathcal{Y}$ represents the domain of images captured from the real world (target). Let $\mathbf{X} \sim P_{\mathbf{X}}$ and $\mathbf{Y} \sim P_{\mathbf{Y}}$ be random objects representing random realizations from these spaces, with distributions $P_{\mathbf{X}}$ and $P_{\mathbf{Y}}$ respectively, where $\mathbf{X} \in \mathcal{X}$ and $\mathbf{Y} \in \mathcal{Y}$. Let $f : \mathcal{X} \to \mathcal{Y}$ denote a mapping function that transforms elements from the source domain to the target domain, and let $g : \mathcal{Y} \to \mathcal{X}$ denote a mapping function for the reverse transformation. These mappings can be implemented as deep neural networks due to their flexibility and capacity for learning complex transformations. However, our study primarily focuses on the forward mapping $f : \mathcal{X} \to \mathcal{Y}$, which transforms elements from the source domain $\mathcal{X}$ to the target domain $\mathcal{Y}$.

The objective of the image-to-image translation problem is to learn (find) these mappings f and g such that: **(i)** the distribution of the mapped object approximates the distribution of the target object, i.e., $P_{f(\mathbf{X})} \approx P_{\mathbf{Y}}$ and/or $P_{\mathbf{X}} \approx P_{g(\mathbf{Y})}$; and **(ii)** the mapping preserves or captures specific characteristics or features of the input images. This objective can be formally expressed as a constraint optimization problem, where the mapped images maintain certain predefined properties or metrics of similarity with the input images, fundamental to tasks like style transfer, domain adaptation, or generative modeling.

Let $\mathsf{dist}\left(P_{f(\mathbf{X})}, P_{\mathbf{Y}}\right)$ denote a discrepancy measure between the distributions of the transformed source images and the target images. For example, one can use the f-divergence $\mathsf{dist}(P_{f(\mathbf{X})}, P_{\mathbf{Y}}) = \mathrm{D}_{\mathsf{f}}(P_{f(\mathbf{X})} \| P_{\mathbf{Y}})$ as such a measure. The optimization problem then aims to minimize a loss function that quantifies both the distributional similarity and the preservation of image characteristics:

$$\min_{f,g} \quad \mathsf{dist}\left(P_{f(\mathbf{X})}, P_{\mathbf{Y}}\right) + \mathsf{dist}(P_{g(\mathbf{Y})}, P_{\mathbf{X}}) + \lambda_{\mathsf{x}} \Phi_{\mathsf{x}}(\mathbf{X}, f(\mathbf{X})) + \lambda_{\mathsf{y}} \Phi_{\mathsf{y}}(\mathbf{Y}, g(\mathbf{Y})), \quad (1)$$

where Φ_{x} and Φ_{y} are penalty functions that enforce the preservation of desired features in the transformed images, with λ_{x} and λ_{y} balancing the importance of distribution similarity and feature preservation.

3.2 Applying General Formulation to Related Works

DECENT [49]: The DECENT objective is introduced as:

$$\min_{f} \quad \mathcal{L}_{\mathsf{gan}} + \lambda_{\mathsf{identity}}\, \mathcal{L}_{\mathsf{identity}} + \lambda_{\mathsf{density}}\, \mathcal{L}_{\mathsf{density}}, \tag{2}$$

where $\mathcal{L}_{\text{gan}} = \mathbb{E}_{P_{\mathbf{X}}}\left[\log(1-D(f(\mathbf{X}))\right] + \mathbb{E}_{P_{\mathbf{Y}}}[\log D(\mathbf{Y})]$, $\mathcal{L}_{\mathsf{identity}} = \mathbb{E}_{P_{\mathbf{Y}}}[f(\mathbf{Y})-\mathbf{Y}]$, and $\mathcal{L}_{\mathsf{density}} = \mathsf{V}\left(\frac{h_{\mathcal{X}}(\mathbf{X})}{h_{\mathcal{Y}}(f(\mathbf{X}))}\right)$, with V as the variance function, $h_{\mathcal{X}}$ and $h_{\mathcal{Y}}$ being density estimators for the corresponding domains, and D as the discriminator (scoring function).

Given our general problem formulation as described in equation (1), it's important to note that in many I2I translation models—particularly those influenced by the CycleGAN framework—the functions f and g work together to enforce cycle consistency. This means that for any image $\mathbf{X} \in \mathcal{X}$, the transformation sequence $\mathbf{X} \to f(\mathbf{X}) \to g(f(\mathbf{X}))$ should closely approximate $\mathbf{X}$. Similarly, this principle applies in reverse, ensuring that the mappings f and g function as approximate inverses of one another. This preserves the content of the images while facilitating translation between domains. Therefore, the identity loss $\mathcal{L}_{\mathsf{identity}}$ is strategically implemented to reinforce this principle by encouraging the function f to act as an identity map when provided with inputs from its target domain $\mathcal{Y}$. Thus, $\mathcal{L}_{\text{gan}}$ corresponds to $\mathsf{dist}\left(P_{f(\mathbf{X})}, P_{\mathbf{Y}}\right)$, $\mathcal{L}_{\mathsf{identity}}$ corresponds to $\Phi_{\mathsf{y}}(\mathbf{Y}, g(\mathbf{Y}))$, and $\mathcal{L}_{\mathsf{density}}$ corresponds to $\Phi_{\mathsf{x}}(\mathbf{X}, f(\mathbf{X}))$ in (1).

VSAIT [42]: The VSAIT objective is introduced as follows:

$$\min_{f} \quad \mathcal{L}_{\mathsf{gan}} + \lambda\, \mathcal{L}_{\mathsf{VSA}}, \tag{3}$$

where $\mathcal{L}_{\mathsf{gan}}$ represents the hypervector adversarial loss, aimed at aligning the distribution of generated images with that of the target images. Meanwhile, $\mathcal{L}_{\mathsf{VSA}}$ is a loss designed to ensure the generator preserves the source content and minimizes semantic flipping.

UNSB [20]: In the context of our general problem formulation, the Schrödinger Bridge for image-to-image translation is tailored to find a mapping $f : \mathcal{X} \to \mathcal{Y}$ that minimizes:

$$\min_{f}\ \mathsf{dist}\left(P_{f(\mathbf{X})}, P_{\mathbf{Y}}\right) + \lambda\, \Phi_{\mathsf{x}}(\mathbf{X}, f(\mathbf{X})), \tag{4}$$

where $\mathsf{dist}\left(P_{f(\mathbf{X})}, P_{\mathbf{Y}}\right) = \mathrm{D}_{\mathsf{KL}}(P_{f(\mathbf{X})} \| P_{\mathbf{Y}})$ is the Kullback-Leibler divergence.

CodeFormer [54]: The objective of CodeFormer is introduced as follows:

$$\min_{f} \quad \mathcal{L}_{\mathsf{L}_1} + \mathcal{L}_{\mathsf{perceptual}} + \mathcal{L}_{\mathsf{code}} + \lambda_{\mathsf{gan}}\, \mathcal{L}_{\mathsf{gan}}, \tag{5}$$

where $\mathcal{L}_{\mathsf{L}_1}$ represents the L_1 loss in the image domain (between source and targeted images), $\mathcal{L}_{\mathsf{perceptual}}$ denotes the L_2 loss in the embedding space (between embeddings of the source and target images), $\mathcal{L}_{\mathsf{code}}$ is the L_2 loss of codeword approximations, and $\mathcal{L}_{\mathsf{gan}}$ is the typical adversarial loss between the source image and the reconstructed image. Considering our general problem formulation (1), the $\mathcal{L}_{\mathsf{L}_1}$ and $\mathcal{L}_{\mathsf{gan}}$ losses contribute towards the $\mathsf{dist}\left(P_{f(\mathbf{X})}, P_{\mathbf{Y}}\right) + \mathsf{dist}(P_{g(\mathbf{Y})}, P_{\mathbf{X}})$ terms, while the other terms act as penalty functions. For more details, we refer the readers to Section 6 of [31], where the authors address generative compression

techniques from the perspective of the transform coding problem and the classical Shannon rate-distortion theorem.

DDIM Inversion [40]: The objective of DDIM Inversion can be aligned with the general problem formulation in equation (1) by introducing an optimization problem that seeks to minimize:

$$\min_{f} \quad \mathsf{dist}\left(P_{f(\mathbf{X})}, P_{\mathbf{Y}}\right) + \lambda_{\mathsf{x}} \Phi_{\mathsf{x}}(\mathbf{X}, f\left(\mathbf{X}\right)). \tag{6}$$

Having outlined the brief theoretical underpinnings and methodological frameworks for enhancing the realism of synthetic 3D imagery, we now proceed to empirically validate these approaches through a series of experiments designed to assess their efficacy in practical applications.

4 Experiments

Methodology: For transferring realism, we began by exploring various methods mentioned in Section 2, namely, CodeFormer [54], VSAIT [42], UNSB [20], Decent [49], DDIM Inversion and StyleGAN Inversion [48]. The goal is to apply Realism Transfer methods to unrealistic images (*i.e.*, $\mathcal{D}_{\mathsf{syn}}$) to get more photo-realistic versions (*i.e.*, $\mathcal{D}_{\mathsf{RT}}$). These versions are used for training and evaluating a FR system. For evaluation, we report verification accuracy (*i.e.*, True Acceptance Rate (TAR)), where the thresholds are set using cross-validation [16] (see Table 2), and TARs at different thresholds determined by fixed False Match Rates (FMR) in Table 3 [26].

Experiment Setup: In the case of the CodeFormer, we utilized the pre-trained models provided by the authors. These models were trained solely on the FFHQ dataset [18], which does not contain any identity labels, as the identity information was not used in their restoration method.

For training unpaired I2I methods, specifically, VSAIT, UNSB, and DECENT, we randomly selected five shards for the source domain (*i.e.*, 3D-rendered human face images), each containing $20,000$ images from the DigiFace1M dataset. Similarly, for the target domain, we randomly selected five shards, each containing $20,000$ images from the FFHQ dataset, and experimented with training these three models using multiple combinations of source and target shards. After training the realism transfer methods, we selected two according to the time they needed to process an image and qualitative examination of the processed images, which are depicted by *Time/Image (s)* and *Qualitative Ex* respectively in Table 1. The processing time was measured on an NVIDIA RTX 3090 Ti across all methods. Figure 3 presents some qualitative results of various methods. As can be qualitatively observed from Figure 3, CodeFormer generally performed very well across all samples, preserving the entire facial structure. In

Table 1: Processing time (*i.e.*, Time/image(s)) and Qualitative Image quality assessment (*i.e.*, Qualitative Ex) of different realism transfer methods.

Metric \ Method	CodeFormer	VSAIT	DECENT	UNSB	DDIM Inversion
Time/Image (s)	0.41	0.015	0.13	0.38	8.7
Qualitative Ex	Good	Average	Average	Average	Good

contrast, VSAIT, DECENT, and UNSB did not consistently produce quality images. Notably, these models sometimes dislocated parts of the images, resulting in multiple eyes and mouths. Surprisingly, as we will demonstrate in the next section, VSAIT boosted the performance of FR systems. Here, 'UNSB-NE-1' and 'UNSB-NE-5' refer to the number of Neural Evaluation (NE) steps of the method; for more details, please refer to the original paper. Finally, images produced by DDIM-Inversion appear as smoothed-out versions of the originals. Among the examined methods, we chose CodeFormer because of its good quality and reasonable compute time and VSAIT for its lower compute time and slightly better quality than other I2I methods for the final FR experiments in the next section.

4.1 Face Recognition Experiments

For a fair comparison between different methods, we trained an FR system consisting of a ResNet50 backbone as modified in ArcFace's implementation [11], with the AdaFace [21] head for contrastive loss. We trained a separate network for each of the methods mentioned in the previous section, namely, the original DigiFace1M, and translated versions of the images generated using CodeFormer and VSAIT. We name the translated dataset **RealDigiFace**. We also included an FR baseline that methods like DCFace and IDiffFace are using; we used the pre-trained model provided by the AdaFace paper, which was trained on the WebFace4M dataset. For FR benchmarking, we considered various datasets including LFW [16], CFPFP [38], CPLFW [52], CALFW [53], AgeDB [28], which consist of high-quality images with various lighting, poses, and ages. We also benchmarked against IJB-C [26], which is amongst the most challenging FR benchmarks in the literature. The results are reported in Table 2 and Table 3.

In the tables mentioned, the first column, *Transfer Method*, refers to the translation method used to translate the dataset. For example, if we want to translate the DigiFace1M dataset using CodeFormer, the *Transfer Method* column for the row corresponding to this experiment is set to CodeFormer. For the case of the WebFace4M, IDiffFace, and DCFace, we did not apply the translation, as expected, since they are not 3D-Rendered data, and we wanted to compare with these datasets as is. The *Type* column refers to the nature of the dataset, which can be either *Real* (collected from the real world), *Syn* (synthetically generated), or *Syn-RT* (translated from a *Syn* dataset using the method mentioned in the *Transfer Method*). The *SynGen Req* columns depict whether the *Transfer Method* or the method used for generating the DigiFace1M, DCFace, and IDiffFace dataset requires the identity labels or a pre-trained FR system. Here, *No-Req* means that neither the translation method nor the method used to generate the original dataset (i.e., DigiFace1M in our experiments) requires the identity label or pre-trained FR system, and *Pre-Trained FR* indicates that generating the dataset required a pre-trained FR system, which is *undesirable* for the problem setting.

We want to emphasize that we repeated the experiments two to four times, reporting the *mean* and *standard deviation* (std) across all benchmarks (*i.e.*, if we observed high variance we repeated the experiment), and also trained the FR

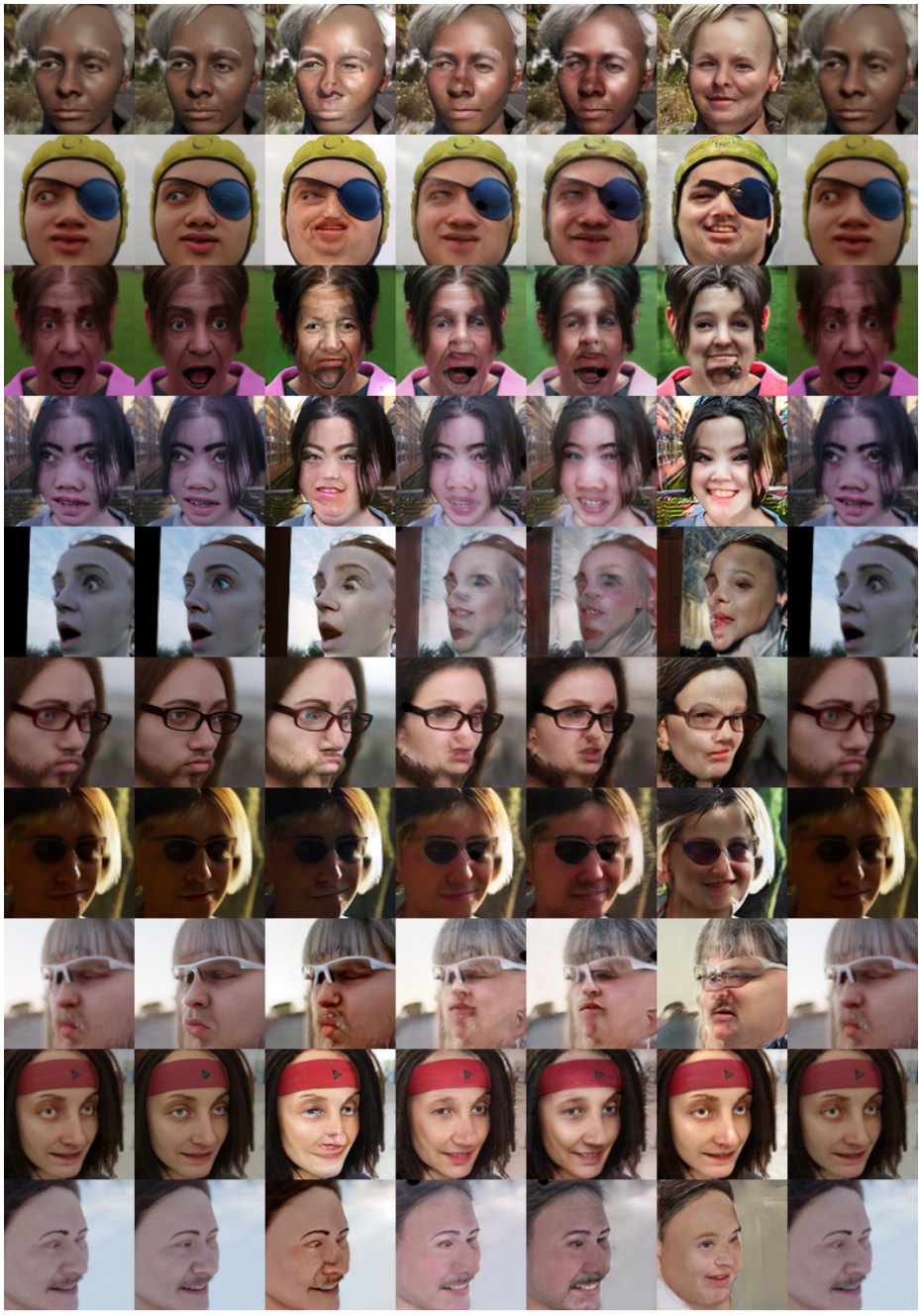

Fig. 3: From the left to right, the first column corresponds to the original DigiFace1M dataset, and the next columns are from after applying different translation tasks to the original images, CodeFormer, VSAIT, DECENT, UNSB-NE-1, UNSB-NE-5 and DDIM Inversion, respectively.

models under the same settings (i.e., all were trained using an IR50 backbone and AdaFace head, with the same early stopping procedure, etc.) for a fair comparison and to ensure that the conclusions drawn are more reliable.

In the case of Table 2, compared to the model trained on the DigiFace1M, we observed an average improvement of 2.0% over the FR model trained on images generated after transferring them using CodeFormer and VSAIT, with CodeFormer demonstrating a slight advantage in all datasets: LFW, CFPFP, CPLFW, CALFW, and AgeDB. However, it can be observed that there is a significant performance gap between the model trained on the WebFace4M and all other methods, including the DCFace and IDiffFace models, which use such a competitive FR system for generating their images.

In the challenging benchmark of IJB-C, as shown in Table 3, we first want to highlight the significant gap in performance between a strong FR system trained on the WebFace4M dataset across all FMR values, and both DCFace and IDiffFace, as well as our models trained on the transferred images. The performance boost observed of Realism images in the IJB-C across different FMR values is larger than that reported in Table 2, with an average gain of about $3-5\%$.

The performance boost of Realism Transfer is notably larger at lower FMR values. Specifically, for models trained on images translated using CodeFormer, the performance approaches that of DCFace and IDiffFace at lower FMRs. Further, we also plotted ROC Curves in the Figure 4, is also emphasizes that models trained on synthetic data are lagging far behind the models that are trained on

Table 2: Results of different synthetic data generation methodologies used to train multiple FR systems evaluated on the LFW, CFPFP, CPLFW, CALFW, and AgeDB, the last column is the average test accuracy over these five datasets. We are reporting *mean* and *std* over multiple runs of experiments in each row.

Transfer Method	Dataset	Type	SynGen Req	LFW	CFPFP	CPLFW	CALFW	AGEDB	Avg
None	WebFace4M	Real	-	**99.78±0.00**	**98.97±0.00**	**94.17±0.00**	**95.98±0.00**	**97.78±0.00**	**97.34±0.00**
None	DigiFace1M	Syn	No-Req	91.29±0.57	88.62±0.69	70.28±0.42	73.38±1.15	68.24±2.17	78.14±0.84
VSAIT [42]	DigiFace1M	Syn-RT	No-Req	92.87±0.15	90.25±0.17	72.91±0.68	75.98±0.28	***70.83±1.22***	80.32±0.25
CodeFormer [54]	DigiFace1M	Syn-RT	No-Req	***93.07±0.27***	***90.50±0.26***	***73.02±0.62***	***76.59±0.19***	70.19±2.57	***80.40±0.29***
None	IDiffFace [8]	Syn	Pre-Trained FR	96.37±0.15	95.54±0.11	73.00±0.47	86.24±0.29	78.29±0.63	84.58±0.16
None	DCFace [22]	Syn	Pre-Trained FR	97.94±0.14	97.87±0.08	78.99±0.49	90.35±0.30	87.46±0.46	89.51±0.13

Table 3: Results of different synthetic data generation methodologies used to train multiple FR systems evaluated on the IJB-C benchmark, here the numbers in the header of the last six columns represent the different TAR@FMR. We are reporting *mean* and *std* over multiple runs of experiments in each row.

Transfer Method	Dataset	Type	SynGen Req	1e-06	1e-05	1e-04	0.001	0.01	0.1
None	WebFace4M	Real	-	**91.78±0.00**	**95.22±0.00**	**96.98±0.00**	**98.14±0.00**	**98.84±0.00**	**99.40±0.00**
None	DigiFace1M	Syn	No-Req	18.80±3.83	28.96±5.16	41.35±5.32	56.38±5.18	72.11±4.30	87.55±2.76
VSAIT [42]	DigiFace1M	Syn-RT	No-Req	20.14±2.98	30.94±2.82	43.98±2.88	59.84±2.12	75.69±1.67	90.21±0.67
CodeFormer [54]	DigiFace1M	Syn-RT	No-Req	***23.72±2.76***	***32.48±3.53***	***45.88±3.53***	***61.74±2.61***	***77.27±1.54***	***90.72±0.81***
None	IDiffFace [8]	Syn	Pre-Trained FR	29.21±3.97	41.77±2.66	56.19±1.51	71.42±0.79	85.59±0.33	95.44±0.02
None	DCFace [22]	Syn	Pre-Trained FR	40.89±0.21	58.58±1.93	72.69±0.53	83.80±0.00	91.97±0.12	97.25±0.02

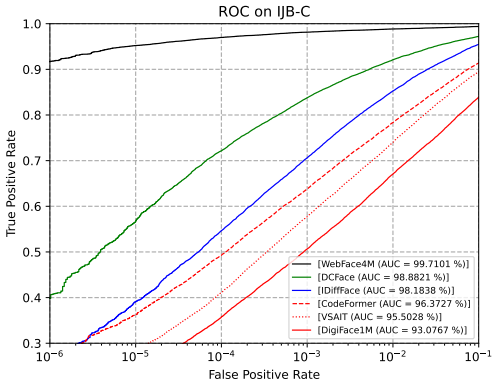

Fig. 4: ROC Curve on the IJB-C benchmark, for each dataset we selected one of the models in which we trained an FR on top of it, and plotted the ROC curve.

real data (*i.e.*, WebFace4M), we can also clearly observe the performance boost of Realism Transfer with respect to the DigiFace1M baseline.

5 Conclusion and Future Work

In this paper, we have explored the potential of utilizing various I2I and face restoration methodologies to address the challenges posed by imperfect rendering in 3D-rendered FR datasets, with the aim of making them more realistic.

Surprisingly, we found that by solely employing transfer models that do not incorporate identity labels in their training paradigm, we can boost the performance of FR systems across all benchmarks—LFW, CFPFF, CPLFW, CALFW, AGEDB, and IJB-C—by 2% to 5%. This improvement is observed in comparison with models trained on the original DigiFace1M dataset, thereby narrowing the performance gap with models that use pre-trained FR data for generating their data. Moreover, this approach moves us closer to our ultimate goal of achieving performance parity with models trained on real data. This opens new avenues for exploring the use of transfer methodologies in the domain of data enhancement for improved downstream model performance. Given that the pipeline for developing a new transfer method and applying it to the entirety of a source dataset is cumbersome and time-consuming—especially since it necessitates multiple trainings of the FR system on the generated data for conclusions—a future work, could be to explore a form of quality assessment metric. This metric would correlate with the final performance of the FR system when trained on the generated dataset, allowing for the evaluation of the transferred data independently. This approach could significantly streamline the process of assessing the potential of newly generated datasets for FR applications.

Acknowledgment This research is based on work conducted in the SAFER project and supported by the Hasler Foundation's Responsible AI program. We would also like to extend our gratitude to Dr. Damien Teney for his thoughtful feedback.

References

1. Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation) (2016), https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:02016R0679-20160504&from=EN
2. Abdal, R., Qin, Y., Wonka, P.: Image2stylegan: How to embed images into the stylegan latent space? In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4432–4441 (2019)
3. Abdal, R., Qin, Y., Wonka, P.: Image2stylegan++: How to edit the embedded images? In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8296–8305 (2020)
4. Anderson, B.D.: Reverse-time diffusion equation models. Stochastic Processes and their Applications **12**(3), 313–326 (1982)
5. Azizi, S., Kornblith, S., Saharia, C., Norouzi, M., Fleet, D.J.: Synthetic data from diffusion models improves imagenet classification. arXiv preprint arXiv:2304.08466 (2023)
6. Bae, G., de La Gorce, M., Baltrušaitis, T., Hewitt, C., Chen, D., Valentin, J., Cipolla, R., Shen, J.: Digiface-1m: 1 million digital face images for face recognition. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 3526–3535 (2023)
7. Baranchuk, D., Rubachev, I., Voynov, A., Khrulkov, V., Babenko, A.: Label-efficient semantic segmentation with diffusion models. arXiv preprint arXiv:2112.03126 (2021)
8. Boutros, F., Grebe, J.H., Kuijper, A., Damer, N.: Idiff-face: Synthetic-based face recognition through fizzy identity-conditioned diffusion model. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19650–19661 (2023)
9. Colbois, L., de Freitas Pereira, T., Marcel, S.: On the use of automatically generated synthetic image datasets for benchmarking face recognition. In: 2021 IEEE International Joint Conference on Biometrics (IJCB). pp. 1–8. IEEE (2021)
10. Coumans, E., Bai, Y.: Pybullet, a python module for physics simulation for games, robotics and machine learning. http://pybullet.org (2016–2021)
11. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4690–4699 (2019)
12. Deng, Y., Yang, J., Chen, D., Wen, F., Tong, X.: Disentangled and controllable face image generation via 3d imitative-contrastive learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5154–5163 (2020)
13. Dinh, T.M., Tran, A.T., Nguyen, R., Hua, B.S.: Hyperinverter: Improving stylegan inversion via hypernetwork. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11389–11398 (2022)
14. Greff, K., Belletti, F., Beyer, L., Doersch, C., Du, Y., Duckworth, D., Fleet, D.J., Gnanapragasam, D., Golemo, F., Herrmann, C., et al.: Kubric: A scalable dataset generator. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3749–3761 (2022)
15. Guo, X., Wu, W., Wang, D., Su, J., Su, H., Gan, W., Huang, J., Yang, Q.: Learning video representations of human motion from synthetic data. In: Proceedings of the

IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20197–20207 (2022)
16. Huang, G.B., Mattar, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In: Workshop on faces in'Real-Life'Images: detection, alignment, and recognition (2008)
17. Karras, T., Aittala, M., Laine, S., Hrknen, E., Hellsten, J., Lehtinen, J., Aila, T.: Alias-free generative adversarial networks. Advances in Neural Information Processing Systems **34**, 852–863 (2021)
18. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401–4410 (2019)
19. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8110–8119 (2020)
20. Kim, B., Kwon, G., Kim, K., Ye, J.C.: Unpaired image-to-image translation via neural schrödinger bridge. In: ICLR (2024)
21. Kim, M., Jain, A.K., Liu, X.: Adaface: Quality adaptive margin for face recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18750–18759 (2022)
22. Kim, M., Liu, F., Jain, A., Liu, X.: Dcface: Synthetic face generation with dual condition diffusion model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12715–12725 (2023)
23. Kleinberg, J., Ludwig, J., Mullainathan, S., Sunstein, C.R.: Discrimination in the Age of Algorithms. Journal of Legal Analysis **10**, 113–174 (04 2019). `https://doi.org/10.1093/jla/laz001`, `https://doi.org/10.1093/jla/laz001`
24. Li, D., Ling, H., Kim, S.W., Kreis, K., Fidler, S., Torralba, A.: Bigdatasetgan: Synthesizing imagenet with pixel-wise annotations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21330–21340 (2022)
25. Ma, J., Bai, S., Zhou, C.: Pretrained diffusion models for unified human motion synthesis. arXiv preprint arXiv:2212.02837 (2022)
26. Maze, B., Adams, J., Duncan, J.A., Kalka, N., Miller, T., Otto, C., Jain, A.K., Niggel, W.T., Anderson, J., Cheney, J., et al.: Iarpa janus benchmark-c: Face dataset and protocol. In: 2018 international conference on biometrics (ICB). pp. 158–165. IEEE (2018)
27. Melzi, P., Rathgeb, C., Tolosana, R., Vera-Rodriguez, R., Lawatsch, D., Domin, F., Schaubert, M.: Gandiffface: Controllable generation of synthetic datasets for face recognition with realistic variations. arXiv preprint arXiv:2305.19962 (2023)
28. Moschoglou, S., Papaioannou, A., Sagonas, C., Deng, J., Kotsia, I., Zafeiriou, S.: Agedb: the first manually collected, in-the-wild age database. In: proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 51–59 (2017)
29. Qiu, H., Yu, B., Gong, D., Li, Z., Liu, W., Tao, D.: Synface: Face recognition with synthetic data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10880–10890 (2021)
30. Rahimi, P., Ecabert, C., Marcel, S.: Toward responsible face datasets: modeling the distribution of a disentangled latent space for sampling face images from demographic groups. arXiv preprint arXiv:2309.08442 (2023)
31. Razeghi, B., Calmon, F.P., Gunduz, D., Voloshynovskiy, S.: Bottlenecks club: Unifying information-theoretic trade-offs among complexity, leakage, and utility. IEEE Transactions on Information Forensics and Security **18**, 2060–2075 (2023)

32. Razeghi, B., Rahimi, P., Marcel, S.: Deep privacy funnel model: From a discriminative to a generative approach with an application to face recognition. arXiv preprint arXiv:2404.02696 (2024)
33. Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., Cohen-Or, D.: Encoding in style: a stylegan encoder for image-to-image translation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2287–2296 (2021)
34. Richter, S.R., AlHaija, H.A., Koltun, V.: Enhancing photorealism enhancement. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(2), 1700–1715 (2022)
35. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
36. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22500–22510 (2023)
37. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al.: Laion-5b: An open large-scale dataset for training next generation image-text models. Advances in Neural Information Processing Systems **35**, 25278–25294 (2022)
38. Sengupta, S., Chen, J.C., Castillo, C., Patel, V.M., Chellappa, R., Jacobs, D.W.: Frontal to profile face verification in the wild. In: 2016 IEEE winter conference on applications of computer vision (WACV). pp. 1–9. IEEE (2016)
39. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: International conference on machine learning. pp. 2256–2265. PMLR (2015)
40. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)
41. Sun, D., Vlasic, D., Herrmann, C., Jampani, V., Krainin, M., Chang, H., Zabih, R., Freeman, W.T., Liu, C.: Autoflow: Learning a better training set for optical flow. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10093–10102 (2021)
42. Theiss, J., Leverett, J., Kim, D., Prakash, A.: Unpaired image translation via vector symbolic architectures. In: European Conference on Computer Vision. pp. 17–32. Springer (2022)
43. Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., Cohen-Or, D.: Designing an encoder for stylegan image manipulation. ACM Transactions on Graphics (TOG) **40**(4), 1–14 (2021)
44. Tritrong, N., Rewatbowornwong, P., Suwajanakorn, S.: Repurposing gans for one-shot semantic part segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4475–4485 (2021)
45. Van Den Oord, A., Vinyals, O., et al.: Neural discrete representation learning. Advances in neural information processing systems **30** (2017)
46. Wood, E., Baltrušaitis, T., Hewitt, C., Dziadzio, S., Cashman, T.J., Shotton, J.: Fake it till you make it: face analysis in the wild using synthetic data alone. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3681–3691 (2021)
47. Wu, Y., Zhang, J., Fu, H., Jin, X.: Lpff: A portrait dataset for face generators across large poses. arXiv preprint arXiv:2303.14407 (2023)

48. Xia, W., Zhang, Y., Yang, Y., Xue, J.H., Zhou, B., Yang, M.H.: Gan inversion: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(3), 3121–3138 (2022)
49. Xie, S., Ho, Q., Zhang, K.: Unsupervised image-to-image translation with density changing regularization. Advances in Neural Information Processing Systems **35**, 28545–28558 (2022)
50. Yang, P., Zhou, S., Tao, Q., Loy, C.C.: PGDiff: Guiding diffusion models for versatile face restoration via partial guidance. In: NeurIPS (2023)
51. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Learning face representation from scratch. arXiv preprint arXiv:1411.7923 (2014)
52. Zheng, T., Deng, W.: Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. Beijing University of Posts and Telecommunications, Tech. Rep **5**(7) (2018)
53. Zheng, T., Deng, W., Hu, J.: Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments. arXiv preprint arXiv:1708.08197 (2017)
54. Zhou, S., Chan, K.C., Li, C., Loy, C.C.: Towards robust blind face restoration with codebook lookup transformer. In: NeurIPS (2022)
55. Zhu, Z., Huang, G., Deng, J., Ye, Y., Huang, J., Chen, X., Zhu, J., Yang, T., Lu, J., Du, D., et al.: Webface260m: A benchmark unveiling the power of million-scale deep face recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10492–10502 (2021)