TCDiff: Triple Condition Diffusion Model with 3D Constraints for Stylizing Synthetic Faces

Bernardo Biesseck*†, Pedro Vidal*, Luiz Coelho‡, Roger Granada‡, David Menotti*
*Depart. of Informatics, Federal University of Paraná, Curitiba, PR, Brazil {bjgbiesseck, pbqv20, menotti}@inf.ufpr.br
†Federal Institute of Mato Grosso (IFMT), Pontes e Lacerda, Brazil {bernardo.biesseck}@ifmt.edu.br
‡unico - idTech, Brazil {luiz.coelho, roger.granada}@unico.io

***Abstract*—A robust face recognition model must be trained using datasets that include a large number of subjects and numerous samples per subject under varying conditions (such as pose, expression, age, noise, and occlusion). Due to ethical and privacy concerns, large-scale real face datasets have been discontinued, such as MS1MV3, and synthetic face generators have been proposed, utilizing GANs and Diffusion Models, such as SYNFace, SFace, DigiFace-1M, IDiff-Face, DCFace, and GANDiffFace, aiming to supply this demand. Some of these methods can produce high-fidelity realistic faces, but with low intra-class variance, while others generate high-variance faces with low identity consistency. In this paper, we propose a Triple Condition Diffusion Model (TCDiff) to improve face style transfer from real to synthetic faces through 2D and 3D facial constraints, enhancing face identity consistency while keeping the necessary high intra-class variance. Face recognition experiments using 1k, 2k, and 5k classes of our new dataset for training outperform state-of-the-art synthetic datasets in real face benchmarks such as LFW, CFP-FP, AgeDB, and BUPT. Our source code is available at: https://github.com/BOVIFOCR/tcdiff.**

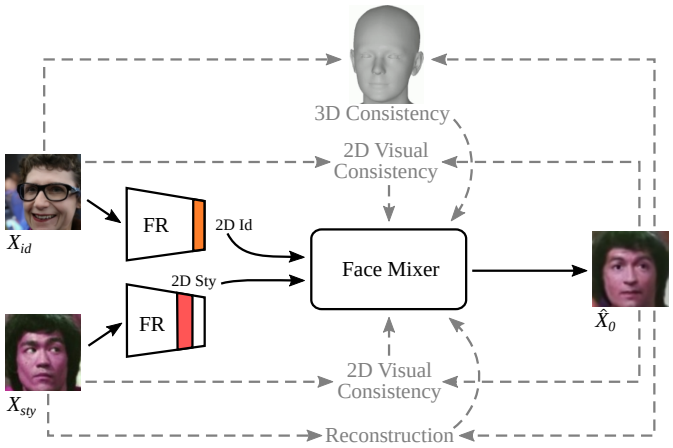

Fig. 1. Overview of the proposed synthetic face mixer TCDiff with 2D and 3D consistency constraints (gray elements). Intermediate style features are extracted from a style image X_{sty} and applied to a synthetic identity image X_{id}, generating a new stylized sample $\hat{X}_0$.

I. Introduction

In recent years, the availability of large face recognition datasets containing thousands of real faces, such as CASIA-WebFace [1], VGGFace2 [2], MS1MV3 [3], WebFace260M [4], and Glint360K [5], has contributed to remarkable advancements in face recognition across various challenging domains, including pose, age, occlusions, and noise. With such data, deep neural networks trained with sophisticated angular margin loss functions, such as SphereFace [6], CosFace [7], ArcFace [8], CurricularFace [9], MagFace [10] and AdaFace [11], have achieved impressive performances on different benchmarks.

However, datasets of this nature present critical ethical, annotation, and bias problems [12]. Furthermore, the long-tailed distribution of samples in many datasets poses additional challenges, necessitating careful network architecture and loss function design to ensure the robustness of model generalization. These challenges also make it difficult to explore facial attribute influences like expression, pose, and illumination. In contrast, learning-based face recognition models encode facial images into fixed-dimensional embedding vectors, enabling various tasks like identification and verification. While publicly available datasets have driven recent progress, they come with associated problems. Synthetic datasets offer a potential solution, providing privacy benefits, virtually unlimited data generation, and control over demographic characteristics. This contrasts with real-world datasets, which are constrained by privacy regulations and representational biases.

Due to such advantages, synthetic faces in face recognition have attracted attention for their potential to mitigate privacy concerns and long-standing dataset biases, such as long-tail distributions and demographic imbalances. Recently, face recognition competitions using synthetic faces have been held, such as FRCSyn [13] [14] and SDFR [15], showing the increasing interest of the research community on this topic.

Generating synthetic faces and manipulating attributes such as pose, expression, age, noise, and occlusion with high visual fidelity and identity consistency is a challenging task due to the ill-posed nature of representing 3D objects in 2D planes. Therefore, 3D facial constraints might improve model learning and reduce facial inconsistencies.

In this paper, we propose a Triple Condition Diffusion Model (TCDiff) to stylize a synthetic identity face with real style attributes from real faces, such as pose, expression, age, noise, occlusion, shadow, hair, etc., with 2D and 3D consistency constraints, aiming to enhance intra-class identity consistency. Fig. 1 presents an overview of our method and the constraints computed with identity image (X_{id}), style image

979-8-3503-7603-6/24/$31.00 © 2024 IEEE

(X_{sty}), and stylized image ($\hat{X}_0$). Our experimental results show that enhancing intra-class identity consistency improves synthetic dataset quality when training face recognition models with few classes.

II. Related Work

Face synthesis has emerged as a prominent area of research, driven by advancements in deep generative models like GANs [16], [17] and Diffusion Models [18]. Such methods excel at generating high-quality facial images with unique identities. However, they often lack the intra-class variance necessary to train powerful face recognition models. In this regard, recent approaches explore such variance in real-face datasets, mixing synthetic and real images to generate multiple samples from the same synthetic subject.

SynFace [19] proposes a Mixup Face Generator designed to create synthetic face images with different identities. To mitigate the domain gap between the synthetic and real face data, the method incorporates a Domain Mixup module regularized by an angular margin loss. In contrast, SFace [20] trains a StyleGAN2-ADA [21] using identity labels as conditional constraints. These constraints are similarly regularized by an angular margin loss.

DigiFace-1M [12] employs the 3DMM-based model FaceSynthetics [22] to generate multiple synthetic faces, varying their expression and pose parameters. The rendering pipeline from FaceSynthetics further enhances flexibility by allowing modifications in the background and illumination settings in the images. After generating original images, data augmentation techniques including flipping, cropping, adding noise, blurring, and warping are applied to improve the face recognition performance. Despite the 3D consistency in images, this dataset has limitations in appearance due to the intrinsic synthetic texture of faces.

GANDiffFace [23] combines the strengths of GANs and Diffusion Models to produce realistic faces while incorporating intra-class variance. Initially, synthetic faces are generated using StyleGAN3 [24] trained on the FFHQ dataset [25], and grouped based on by extracted facial attributes such as pose, expression, illumination, gender, and race. Support Vector Machine (SVM) classifiers are trained to distinguish each group. The normal vectors concerning the resulting separation hyperplanes are used as directions to edit facial attributes of faces in latent space. Despite achieving realistic appearances for the generated synthetic faces, there remains a gap in the evaluation performance using real face testing benchmarks of face recognition models, trained on their synthetic dataset.

IDiffFace [26] uses a Diffusion Model trained on the FFHQ [25] real face dataset to create new synthetic faces. These faces are generated using a U-Net-based architecture enriched with residual and attention blocks to encourage the model to improve the intra-class variance generation ability. To prevent overfitting and ensure diverse outputs, they also propose a Contextual Partial Dropout (CPD) technique.

DCFace [27] introduces an approach to minimize the distribution gap between synthetic and real face datasets by employing a diffusion model. This model integrates visual constraints to transfer the stylistic characteristics of real faces onto synthetic faces, thereby enhancing the intra-class variance. Initially trained on the CASIA-WebFace [1] real face dataset, the model utilizes statistics derived from intermediate features of images, assuming these contain style information to be transferred to any other identity. Despite these efforts, some artifacts and identity inconsistencies persist in the synthetic output.

Fig. 2. Face samples of real (first row) and synthetic datasets. SYNFace has low identity consistency and low intra-class variance; SFace has low identity consistency; DigiFace-1M, GANDiffFace, and IDiffFace have high identity consistency but low intra-class variance; DCFace has low identity consistency and high intra-class variance.

Fig. 2 illustrates some samples generated by the aforementioned methods. Each row corresponds to samples of distinct synthetic subjects, while columns contain different samples of the same subject. While the limited number of images may not be sufficient to provide an accurate visual analysis, they offer some initial intuitions about their characteristics, such as intra-class variance and identity consistency.

III. Face Style Transfer

Image style transfer is a technique that generates novel images by merging the content of one image with the stylistic elements of another [28]. This process leverages Convolutional Neural Networks (CNNs) pre-trained on image classification tasks to extract hierarchical intermediate feature representations. The content of an image is captured by the higher layers of the network, which encode the image's structure and objects, while the style is represented by the correlations between feature maps in the lower layers, known as Gram matrices. To achieve style transfer, the technique minimizes a loss function that combines the content loss, which measures the difference between content representations of the original and generated images, and style loss, which quantifies the difference between style representations of the original and generated images. By iteratively adjusting a white noise image

based in this combined loss function, the network gradually synthesizes the final output that maintains the content of one image while adopting the style of another.

In the face recognition field, a person's identity can be expressed mainly by facial parts such as eyes, nose, lips, eyebrows, etc., and their spatial position in the face. Meanwhile, style is related to facial pose, expression, age, noise, occlusion, color, etc [29]. Achieving a perfect disentanglement of identity and style representation remains a significant challenge in deep learning. Existing style transfer methods aim to manage this tradeoff depending on the final goal [30].

IV. Proposed Approach

For face style extraction, we adopt the proposed model E_{sty} of DCFace [27], which uses intermediate feature maps $I_{sty} \in \mathbb{R}^{C\times H\times W}$ extracted with a pre-trained and fixed weights face recognition model F_s from a given input face image X_{sty}, where C, H and W are the number of channels, height, and width of the feature maps, respectively. Each feature map is divided into a grid $k \times k$ and each element $I^{k_i}_{sty} \in \mathbb{R}^{C\times \frac{H}{k}\times \frac{W}{k}}$ is mapped on the mean and variance of $I^{k_i}_{sty}$ as

$$\hat{I}^{k_i} = \text{BN}(\text{Conv}(\text{ReLU}(\text{Dropout}(I^{k_i}_{sty})))), \tag{1}$$

$$\mu^{k_i}_{sty} = \text{SpatialMean}(\hat{I}^{k_i}), \quad \sigma^{k_i}_{sty} = \text{SpatialStd}(\hat{I}^{k_i}), \tag{2}$$

$$s^{k_i} = \text{LN}((W_1 \odot \mu^{k_i}_{sty} + W_2 \odot \sigma^{k_i}_{sty}) + P_{emb}), \tag{3}$$

$$E_{sty}(X_{sty}) := s = [s^1, s^2, s^{k_i}, ..., s^{k\times k}, s'], \tag{4}$$

where s' corresponds to $\hat{I}^{k_i}_{sty}$ being a global feature, where $k = 1$. $P_{emb} \in \mathbb{R}^{50\times C}$ is a learned position embedding [31] that makes the model learn to extract patches styles according to their locations in X_{sty} . BN and LN are BatchNorm [32] and LayerNorm [33] operations.

The face style embedding E_{sty} extracted from X_{sty} is then applied to an identity image X_{id} using a U-Net denoising diffusion probabilistic model (DDPM) [18] face mixer $\epsilon_\theta(X_t, t, E_{id}(X_{id}), E_{sty}(X_{sty}))$, whose architecture is illustrated in Fig. 3. X_t is a noisy version of X_{sty} at time-step t and E_{id} is a face recognition model ResNet50 [34] responsible for extracting a discriminant identity embedding.

Given an identity image X_{id} and a style image X_{sty}, a new stylized image $\hat{X}_0$ is obtained as

$$\hat{X}_0 = (X_t - \sqrt{1-\bar{\alpha}}\epsilon_\theta(X_t, t, X_{id}, X_{sty}))/\sqrt{\bar{\alpha}_t}. \tag{5}$$

where $\bar{\alpha}_t$ is a pre-set variance scheduling scalar [18].

To train the face mixer ϵ_θ, we first employ the mean squared error (MSE) loss L_{MSE} between style image X_{sty} and stylized image $\hat{X}_0$

$$L_{MSE} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\Big(X_{sty}(i,j) - \hat{X}_0(i,j)\Big)^2 \tag{6}$$

to enforce the model to preserve relevant style features of X_{sty} in $\hat{X}_0$. Additionally, to balance identity and style features of

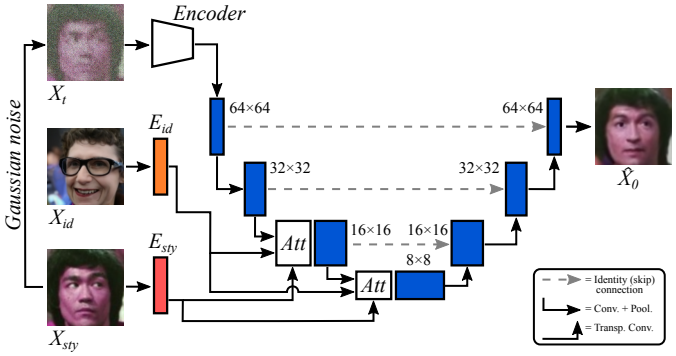

Fig. 3. Simplified illustration of internal architecture of the U-Net DDPM face mixer ϵ_θ. Intermediate feature maps (blue boxes) are extracted from X_t, a noisy version of X_{sty}, by encoders and copied to their corresponding upscale layer. Identity features E_{id}, extracted from X_{id}, and style features E_{sty}, extracted from X_{sty}, are fed to attention modules together with intermediate feature maps, allowing the model learn how to denoise X_t with features from X_{id} and X_{sty}.

X_{id} and X_{sty} in $\hat{X}_0$, we also employ an identity loss L_{ID} through cosine similarity (CS)

$$\begin{aligned} L_{ID} = &-\gamma_t \text{CS}(F(X_{id}), F(\hat{X}_0)) \\ &-(1-\gamma_t)\text{CS}(F(X_{sty}), F(\hat{X}_0)) \end{aligned} \tag{7}$$

where $\gamma_t \in \mathbb{R} \mid 0 \leq \gamma_t \leq 1$.

To enhance intra-class identity consistency when stylizing synthetic faces, we also propose to add a 3D facial shape loss L_{3D}

$$L_{3D} = \sqrt{\sum (x^{3D}_{id} - \hat{x}^{3D}_0)^2} \tag{8}$$

which computes the Euclidean Distance between the 3DMM [35] shape feature vectors x^{3D}_{id} and $\hat{x}^{3D}_0$, obtained from X_{id} and $\hat{X}_0$. Due to the lack of large datasets containing both 2D and 3D scanned representations of real facial, we obtained 3D Morphable Model (3DMM) coefficients during training using the 3D face reconstruction model MICA [36].

The shape of a face in a 3DMM representation is described by the positions of a set of 3D vertices $S = (x_1, y_1, z_1, x_2, y_2, z_2, \ldots, x_n, y_n, z_n)^T \in \mathbb{R}^{3n}$. These vertices form a mesh that captures the geometric structure of the face. Mathematically, the shape vector S can be expressed as a linear combination of a mean shape $\bar{S}$ and a set of shape basis vectors S_i:

$$S = \bar{S} + \sum_{i=1}^{n}\alpha_i S_i, \tag{9}$$

where α_i are the shape coefficients that determine how much each basis vector S_i contributes to the final shape. The shape vector S consists of the 3D coordinates of all vertices in the mesh.

Similarly, the face texture is described by a vector $T = (R_1, G_1, B_1, R_2, G_2, B_2, \ldots, R_n, G_n, B_n)^T \in \mathbb{R}^{3n}$, which is a linear combination of a mean texture $\bar{T}$ and a set of texture basis vectors T_i:

$$T = \bar{T} + \sum_{i=1}^{m} \beta_i T_i, \tag{10}$$

where β_i are the texture coefficients that control the contribution of each texture basis vector T_i. The texture vector T consists of the RGB color values for each vertex in the mesh.

The basis vectors S_i and T_i, for shape and texture, are derived from a Principal Component Analysis (PCA) over a set of real 3D face scans, which allows representing new faces within the training set variance.

To obtain the 3DMM facial shape coefficients x_{id}^{3D} and $\hat{x}_0^{\text{3D}}$ from identity face X_{id} and stylized face $\hat{X}_0$, the MICA [36] method uses the face embedding produced by a state-of-the-art 2D face recognition network [37] as input to a small mapping network $z = M(ArcFace(I)) \in \mathbb{R}^{300}$. Therefore, $x_{id}^{\text{3D}} = M(X_{id})$ and $\hat{x}_0^{\text{3D}} = M(\hat{X}_0)$.

Finally, our total loss function L_T is defined as

$$L_T = L_{MSE} + \lambda_{id} L_{ID} + \lambda_{\text{3D}} L_{\text{3D}}, \tag{11}$$

where λ_{id} and λ_{3D} are scaling parameters to balance the importance of 2D and 3D facial identity constraints.

V. Experiments

This section presents our experimental setup, datasets, obtained results, and qualitative analysis. To fairly evaluate the robustness of our proposed TCDiff face mixer, we adopted the same protocol of DCFace [27] by using the real faces dataset CASIA-WebFace [1] as the training set and a grid 5×5 for style feature extraction. Our model was trained for 10 epochs with batch=16 using AdamW Optimizer [38] with the learning rate of $1\text{e}{-4}$ on one GPU NVIDIA GeForce RTX 3090. We set $\lambda_{\text{ID}} = 0.05$ and varied $\lambda_{\text{3D}} = \{0.001, 0.005, 0.01, 0.05\}$ to analyse the impact of 3D consistency constraints.

After training TCDiff, we selected the same 10k distinct synthetic identities of DCFace [27], which were generated using the publicly released unconditional DDPM [39] trained on FFHQ [40]. Each synthetic identity was stylized with 50 randomly chosen real faces of CASIA-WebFace [1], resulting in a new synthetic dataset of 500k images. Fig. 4 shows 1 synthetic face image X_{id}, 16 real faces X_{sty} from CASIA-WebFace, and their corresponding 16 new samples of X_{id}.

We choose ResNet50 [34] backbone and ArcFace [8] loss as Face Recognition (FR) model to evaluate the quality of the proposed synthetic dataset in cross-dataset scenarios using seven different datasets in face verification (1:1) task: LFW [41] CFP-FF [42], CPLFW [43], CFP-FP [42], AgeDB [44], CALFW [45], and BUPT-CBFace [46]. These datasets are commonly applied in FR to validate or test models. Each dataset contains a verification protocol consisting of face pairs labeled as *genuine* (same person) or *impostor* (different

Fig. 4. Stylized synthetic face samples generated with DCFace [27] and our proposed TCDiff face mixer. The first row shows the original synthetic X_{id} face and 16 real style faces X_{sty} used to create new samples. In our experiments, $\lambda_{\text{3D}} = 0.001$ is the best value to balance intra-class identity consistency and variance.

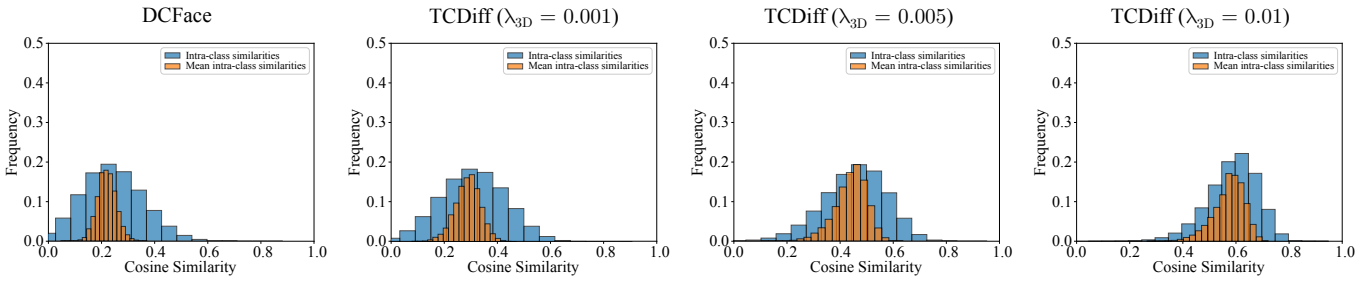

Fig. 5. Intra-class cosine similarities of synthetic datasets generated by DCFace [27] and our proposed model TCDiff with different values of λ_{3D}, computed with ResNet100/Arcface [8] trained on MS1MV3 [3]. The higher the λ_{3D} value, the higher the intra-class identity consistency.

TABLE I
FACE VERIFICATION RESULTS (%) OF RESNET50 ON LFW [41], CFP-FF [42], CPLFW [43], CFP-FP [42], AGEDB [44], CALFW [45] AND BUPT-CBFACE [46] WHEN TRAINING ON DIFFERENT SYNTHETIC FACE DATASETS GENERATED WITH DCFACE [27] AND OUR PROPOSED MODEL TCDIFF. THE BEST RESULTS ARE IN BOLD.

#Train Subj.	Train Dataset	LFW	CFP-FF	CPLFW	CFP-FP	AgeDB	CALFW	BUPT	Avg	Std	Avg-Std
1k	DCFace	88.30	88.70	61.00	62.40	70.10	76.60	75.30	74.63	**11.14**	63.49
	TCDiff ($\lambda_{3D} = 0.001$)	90.50	91.30	62.50	64.20	72.40	80.20	80.30	77.34	11.56	65.78
	TCDiff ($\lambda_{3D} = 0.005$)	**90.55**	**91.87**	**63.83**	**64.42**	**73.10**	**81.40**	**80.67**	**77.98**	11.39	**66.59**
	TCDiff ($\lambda_{3D} = 0.01$)	88.05	89.47	58.71	59.42	67.38	78.61	78.05	74.24	12.68	61.56
2k	DCFace	92.80	94.07	67.48	71.53	77.07	83.55	82.26	81.25	**10.05**	71.20
	TCDiff ($\lambda_{3D} = 0.001$)	**94.22**	**95.07**	**68.18**	**72.03**	**79.13**	**85.03**	**84.14**	**82.54**	10.25	**72.29**
	TCDiff ($\lambda_{3D} = 0.005$)	93.15	94.22	67.05	67.08	76.70	84.26	83.41	80.84	11.15	69.69
	TCDiff ($\lambda_{3D} = 0.01$)	90.70	92.11	61.84	61.08	71.63	81.81	81.25	77.20	12.71	64.49
5k	DCFace	96.40	96.70	72.00	79.10	82.60	87.30	86.40	85.79	8.94	76.85
	TCDiff ($\lambda_{3D} = 0.001$)	**96.87**	**97.26**	**73.92**	**79.86**	**83.68**	**87.92**	**89.18**	**86.96**	**8.58**	**78.38**
	TCDiff ($\lambda_{3D} = 0.005$)	94.30	95.11	68.40	69.60	78.60	86.20	84.60	82.40	10.78	71.63
	TCDiff ($\lambda_{3D} = 0.01$)	92.31	93.74	64.75	63.81	74.96	84.01	83.13	79.53	12.14	67.39
10k	DCFace	**98.02**	97.73	**79.62**	**85.01**	**88.82**	**90.48**	**91.35**	**90.15**	**6.58**	**83.56**
	TCDiff ($\lambda_{3D} = 0.001$)	97.77	**98.04**	78.13	82.21	86.35	90.33	90.73	89.08	7.47	81.61
	TCDiff ($\lambda_{3D} = 0.005$)	95.98	96.69	70.95	71.83	81.13	87.72	87.66	84.57	10.46	74.11
	TCDiff ($\lambda_{3D} = 0.01$)	93.51	95.10	64.95	64.64	74.68	85.01	84.20	80.30	12.54	67.76

person).

LFW (6k pairs) and CFP-FF (7k pairs) protocols are mainly focused on frontal face verification, representing controlled scenarios. Otherwise, CPLFW (6k pairs) and CFP-FP (7k pairs) contain faces with more varied poses to simulate in-the-wild scenarios. AgeDB (7k pairs) and CALFW (6k pairs) focus on comparing faces with large age differences, while BUPT-CBFace (8k pairs) contains the same number of pairs of 4 distinct ethnic groups: Asian, Caucasian, African, and Indian. All protocols were split into 10-fold containing 50% of genuine and 50% of impostor pairs. Following the cross-validation method, we use 9 folds to select the best threshold and 1 for the final test.

VI. DISCUSSION

In this section, we present a qualitative and quantitative analysis of the results obtained with the synthetic datasets generated with DCFace [27] and our proposed model TCDiff for the face recognition task.

Even with few samples, we can visually observe in Fig. 4 that synthetic faces stylized by DCFace [27] have low intra-class consistency, compared to its corresponding original synthetic identity X_{id}. For instance, the stylized male faces 0, 8, and 10 seem to belong to distinct identities. The same happens with the stylized female faces 2 and 9. Otherwise, our face mixer ϵ_θ tends to preserve identity regions, such as eyes, nose, and mouth of original X_{id} in stylized faces to enhance the intra-class consistency imposed by L_{3D} when λ_{3D} varies from 0.001 to 0.01. Stylized faces are completely degraded when $\lambda_{3D} = 0.05$ and this setting was ignored in our experiments.

Such a qualitative analysis is quantitatively confirmed in Fig. 5, where intra-class cosine similarities are presented. Blue bars show the distributions of all $\sum_{i=0}^{M} C_2^{N_i}$ similarities, where M is the number of classes and N_i is the number of samples of the $i\text{-}th$ class, while orange bars show the distributions of the mean intra-class similarities. One can observe higher similarities in faces stylized with our model TCDiff, indicating a higher intra-class consistency due to the shape of eyes, nose, lips, skin color, pose, expression, and facial accessories.

The quality of a face dataset for FR task is assessed not just by the images themselves, but by the performances of FR models trained on it. Results in Table I show that enhancing intra-class consistency improves synthetic datasets quality when training with 1k, 2k, and 5k classes. We hypothesize that such an identity consistency improves inter-class separability when the number of distinct identities are low.

However, this improvement is surpassed when training with 10k classes, indicating that the inter-class variability is also an important property of synthetic datasets for face recognition. ResNet50 performed slightly better on dataset CFP-FF [42] when training with stylized images by our model TCDiff ($\lambda_{3D} = 0.001$), due to the greater existence of frontal faces in such a dataset.

VII. CONCLUSION AND FUTURE WORK

In this work we propose TCDiff, a face style transfer trained with 2D and 3D facial constraints, aiming to improve the quality of synthetic datasets for face recognition. By increasing the importance of 3D constraints, our model can preserve identity features of the input synthetic face to be stylized, which enhances intra-class identity consistency.

This behavior contributes to increasing the quality of small synthetic datasets and might be explored in the future for more classes, as the interest in this field is growing due to the advantages of synthetic data. As a future step, facial expression and pose constraints might be added to the face styler model, aiming to balance better the importance of identity and style features in newly generated samples.

ACKNOWLEDGMENT

This work was supported by a tripartite-contract, i.e., unico - idTech, UFPR (Federal University of Paraná), and FUNPAR

(Fundação da Universidade Federal do Paraná). We thank the Federal Institute of Mato Grosso (IFMT), Pontes e Lacerda, for supporting Bernardo Biesseck, and also thank the National Council for Scientific and Technological Development (CNPq) (# 315409/2023-1) for supporting Prof. David Menotti.

REFERENCES

[1] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," 2014.

[2] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *British Machine Vision Conference*, 2015.

[3] J. Deng, J. Guo, D. Zhang, Y. Deng, X. Lu, and S. Shi, "Lightweight face recognition challenge," in *IEEE CVPR (ICCV) Workshops*, 2019.

[4] Z. Zhu *et al.*, "Webface260m: A benchmark unveiling the power of million-scale deep face recognition," in *IEEE CVPR*, 2021.

[5] X. An, J. Deng, J. Guo, Z. Feng, X. Zhu, Y. Jing, and L. Tongliang, "Killing two birds with one stone: Efficient and robust training of face recognition cnns by partial fc," in *IEEE CVPR*, 2022.

[6] W. Liu *et al.*, "Sphereface: Deep hypersphere embedding for face recognition," in *IEEE CVPR*, 2017.

[7] H. Wang *et al.*, "Cosface: Large margin cosine loss for deep face recognition," in *IEEE CVPR*, 2018.

[8] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," *IEEE CVPR*, 2019.

[9] Y. Huang, Y. Wang, Y. Tai, X. Liu, P. Shen, S. Li, J. Li, F. Huang, and Y. Lab, "Curricularface: Adaptive curriculum learning loss for deep face recognition," in *IEEE CVPR*, 2020.

[10] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "Magface: A universal representation for face recognition and quality assessment," in *IEEE CVPR*, 2021.

[11] M. Kim, A. K. Jain, and X. Liu, "Adaface: Quality adaptive margin for face recognition," in *IEEE CVPR*, 2022.

[12] G. Bae, M. de La Gorce, T. Baltrušaitis, C. Hewitt, D. Chen, J. Valentin, R. Cipolla, and J. Shen, "Digiface-1m: 1 million digital face images for face recognition," in *2023 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2023.

[13] P. Melzi, R. Tolosana, R. Vera-Rodriguez, M. Kim, C. Rathgeb, X. Liu, I. DeAndres-Tame, A. Morales, J. Fierrez, J. Ortega-Garcia, W. Zhao, X. Zhu, Z. Yan, X.-Y. Zhang, J. Wu, Z. Lei, S. Tripathi, M. Kothari, M. H. Zama, D. Deb, B. Biesseck, P. Vidal, R. Granada, G. Fickel, G. Führ, D. Menotti, A. Unnervik, A. George, C. Ecabert, H. O. Shahreza, P. Rahimi, S. Marcel, I. Sarridis, C. Koutlis, G. Baltsou, S. Papadopoulos, C. Diou, N. Di Domenico, G. Borghi, L. Pellegrini, E. Mas-Candela, A. Sánchez-Pérez, A. Atzori, F. Boutros, N. Damer, G. Fenu, and M. Marras, "Frcsyn challenge at wacv 2024: Face recognition challenge in the era of synthetic data," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, January 2024, pp. 892–901.

[14] R. Tolosana, P. Melzi, I. Andres, R. Vera-Rodriguez, M. Kim, C. Rathgeb, X. Liu, A. Morales, J. Fierrez, and J. Ortega-Garcia, "2nd edition frcsyn: Face recognition challenge in the era of synthetic data," 2024.

[15] H. O. Shahreza, C. Ecabert, A. George, A. Unnervik, S. Marcel, N. D. Domenico, G. Borghi, D. Maltoni, F. Boutros, J. Vogel, N. Damer, Ángela Sánchez-Pérez, EnriqueMas-Candela, J. Calvo-Zaragoza, B. Biesseck, P. Vidal, R. Granada, D. Menotti, I. DeAndres-Tame, S. M. L. Cava, S. Concas, P. Melzi, R. Tolosana, R. Vera-Rodriguez, G. Perelli, G. Orrù, G. L. Marcialis, and J. Fierrez, "Sdfr: Synthetic data for face recognition competition," 2024.

[16] Y. Shen, P. Luo, J. Yan, X. Wang, and X. Tang, "Faceid-gan: Learning a symmetry three-player gan for identity-preserving face synthesis," in *IEEE CVPR*, 2018.

[17] Y. Deng, J. Yang, D. Chen, F. Wen, and X. Tong, "Disentangled and controllable face image generation via 3d imitative-contrastive learning," in *IEEE CVPR*, 2020.

[18] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *NeurIPS*. Curran Associates, Inc., 2020.

[19] H. Qiu, B. Yu, D. Gong, Z. Li, W. Liu, and D. Tao, "Synface: Face recognition with synthetic data," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society, 2021.

[20] F. Boutros, M. Huber, P. Siebke, T. Rieber, and N. Damer, "Sface: Privacy-friendly and accurate face recognition using synthetic data," in *IEEE IJCB*, 2022.

[21] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," in *NeurIPS*. Curran Associates, Inc., 2020.

[22] E. Wood, T. Baltrušaitis, C. Hewitt, S. Dziadzio, T. J. Cashman, and J. Shotton, "Fake it till you make it: face analysis in the wild using synthetic data alone," in *IEEE CVPR*, 2021.

[23] P. Melzi, C. Rathgeb, R. Tolosana, R. Vera-Rodriguez, D. Lawatsch, F. Domin, and M. Schaubert, "Gandiffface: Controllable generation of synthetic datasets for face recognition with realistic variations," in *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. Los Alamitos, CA, USA: IEEE Computer Society, oct 2023.

[24] Y. Alaluf, O. Patashnik, Z. Wu, A. Zamir, E. Shechtman, D. Lischinski, and D. Cohen-Or, "Third time's the charm? image and video editing with stylegan3," 2022.

[25] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," *2019 IEEE CVPR*, pp. 4396–4405, 2018.

[26] F. Boutros, J. H. Grebe, A. Kuijper, and N. Damer, "Idiff-face: Synthetic-based face recognition through fizzy identity-conditioned diffusion models," in *IEEE CVPR (ICCV)*, October 2023.

[27] M. Kim, F. Liu, A. Jain, and X. Liu, "Dcface: Synthetic face generation with dual condition diffusion model," in *2023 IEEE CVPR*. IEEE Computer Society, 2023.

[28] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *2016 IEEE CVPR*, 2016.

[29] A. Suwała, B. Wójcik, M. Proszewska, J. Tabor, P. Spurek, and M. Śmieja, "Face identity-aware disentanglement in stylegan," in *IEEE WACV*, 2024.

[30] D. Chen, L. Yuan, J. Liao, N. Yu, and G. Hua, "Explicit filterbank learning for neural image style transfer and image processing," *IEEE TPAMI*, 2021.

[31] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *ICML*. PMLR, 2017.

[32] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015.

[33] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," 2016.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE CVPR*, 2016, pp. 770–778.

[35] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," in *26th Annual Conference on Computer Graphics and Interactive Techniques*, 1999.

[36] W. Zielonka, T. Bolkart, and J. Thies, "Towards metrical reconstruction of human faces," in *ECCV*. Springer International Publishing, 2022.

[37] J. Deng, J. Guo, T. Liu, M. Gong, and S. Zafeiriou, "Sub-center arcface: Boosting face recognition by large-scale noisy web faces," in *ECCV*, 2020.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[39] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *NeurIPS*, 2020.

[40] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," *IEEE TPAMI*, 2021.

[41] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.

[42] S. Sengupta, J. Cheng, C. Castillo, V. Patel, R. Chellappa, and D. Jacobs, "Frontal to profile face verification in the wild," in *IEEE Conference on Applications of Computer Vision*, February 2016.

[43] T. Zheng and W. Deng, "Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments," Beijing University of Posts and Telecommunications, Tech. Rep., 2018.

[44] S. Moschoglou *et al.*, "Agedb: the first manually collected, in-the-wild age database," in *IEEE CVPR Workshop*, 2017, p. 5.

[45] T. Zheng, W. Deng, and J. Hu, "Cross-age LFW: A database for studying cross-age face recognition in unconstrained environments," *CoRR*, vol. abs/1708.08197, 2017.

[46] Y. Zhang and W. Deng, "Class-balanced training for deep face recognition," in *IEEE CVPR Workshops*, 2020.