Controllable Inversion of Black-Box Face Recognition Models via Diffusion: Supplementary Material

Manuel Kansy[1,2] *, Anton Raël[1], Graziana Mignone[2], Jacek Naruniec[2], Christopher Schroers[2], Markus Gross[1,2], and Romann M. Weber[2]

[1]ETH Zurich, Switzerland, [2]DisneyResearch|Studios, Switzerland

{mkansy, grossm}@inf.ethz.ch, anrael@student.ethz.ch, {<first>.<last>}@disneyresearch.com

1. Additional implementation details

ID vectors Table 1 lists the face recognition methods used in this work. Note that we use two implementations for ArcFace [5] and FaceNet [28], one for training and the other one for evaluation in each case. The methods used for training were chosen to match those used in Gaussian sampling [24] and StyleGAN search [35] respectively. The methods used for evaluation were chosen to match the verification accuracy on real images as closely as possible to the values shown in Vec2Face [7] to enable a fair comparison. For both evaluation methods, we extract the identity embeddings for each image as well as its horizontally flipped version and then calculate the angular distance of the concatenated identity embeddings after subtracting the mean embedding, similar to [27]. In order to avoid having the face detector stage of different face recognition vectors influence the qualitative results, we manually confirmed that all shown test images were properly aligned.

Method	Usage	Alignment	Implementation	Checkpoint
AdaFace [15]	Training [1]	Provided MTCNN [37]	Official GitHub repository	"adaface_ir50_ms1mv2.ckpt"
ArcFace [5, 2]	Evaluation	RetinaFace [4] from [3]	Official GitHub repository	"ms1mv3_arcface_r100_fp16"
ArcFace [5, 24]	Training [1]	MTCNN [37] from [8]	From Razzhigaev *et al*. [24]	"torchtest.pt"
FaceNet [28, 27]	Evaluation	Provided MTCNN [37]	From David Sandberg [27]	"20180402-114759"
FaceNet [28, 8]	Training	Provided MTCNN [37]	From Tim Esler [8]	"20180402-114759"
FROM [23]	Training [1]	MTCNN [37] from [8]	Official GitHub repository	"model_p5_w1_9938_9470_6503.pth.tar"
InsightFace [3]	Training	Provided RetinaFace [4]	InsightFace repository [3]	"buffalo_l"

Table 1: Overview over the considered face recognition methods. [1] Only used in supplementary material.

Model We use the official U-net [25] implementation by Dhariwal and Nichol [6, 21] and their recommended hyperparameters, whenever applicable, for the main 64×64 ID-conditioned face generation model and the $64 \rightarrow 256$ super-resolution model as listed in Tab. 2. The U-net architecture is divided into several levels, with each level composed of ResNet [10] blocks and down- or upsampling layers. The U-net also contains global attention layers at 32×32, 16×16, and 8×8 resolutions. The time step t is passed through a sinusoidal position embedding layer, known from transformers [34], and is then added to the residual connection of the ResNet blocks. The most important additions to the baseline model are the identity conditioning module (identity_cond) and introducing classifier-free guidance (classifier_free) by setting the conditioning vector to the 0-vector[1] for 10% of the training samples to obtain an unconditional and conditional setting with just one trained model.

*Corresponding author.

[1]For attribute conditioning, the -1-vector is used since the 0-vector is a valid attribute vector (*e.g.* age 0) and the -1-vector performed better empirically.

Training The training set is composed of images along with their corresponding (pre-computed) ID vectors. We train the 64×64 ID-conditioned face generation model for 100000 batches and the 64×64 unconditional upsampling model for 50000 batches, both with a batch size of 64, learning rate of 10^{-4}, and from scratch. We use the weights with an exponential moving average rate of 0.9999 because it generally leads to better results. Training takes around two days on one NVIDIA RTX 3090 GPU.

Inference All models are trained with $T = 1000$ but respaced to 250 time steps during inference for computational reasons with a negligible decrease in quality. We use a classifier-free guidance scale of 2 unless otherwise stated. Furthermore, we fix the randomness seeds whenever comparing different methods to ensure a fair comparison. Inference (main model + super-resolution to 256×256 resolution) takes around 15 seconds per image when using batches of 16 images on one NVIDIA RTX 3090 GPU. The inference time can be drastically reduced by using fewer respacing steps at a slight decrease in quality, as shown in Fig. 1. For example, when using 10 respacing steps, the inference time decreases to around 1 second per image with a comparable identity fidelity and only slightly fewer details (especially in the background).

	64×64 main model	$64 \times 64 \rightarrow 256 \times 256$ super-resolution model
Diffusion parameters		
diffusion_steps	1000	1000
noise_schedule	cosine	linear
Model parameters		
attention_resolutions	$32, 16, 8$	$32, 16, 8$
classifier_free	True	False
dropout	0.1	0
identity_cond	True	False
learn_sigma	True	True
num_channels	192	192
num_heads	3	4
num_res_blocks	3	2
resblock_updown	True	True
use_fp16	True	True
use_new_attention_order	True	False
use_scale_shift_norm	True	True
Training parameters		
batch_size	64	64
ema_rate	0.9999	0.9999
lr (learning rate)	10^{-4}	10^{-4}
total_steps (batches)	100000	50000

Table 2: Hyperparameters of our diffusion models. We use one diffusion model to generate 64×64 resolution images and one super-resolution diffusion model to increase the resolution to 256×256. All other parameters are named as in the baseline implementation (where applicable).

Time (s)	0.2	0.2	0.3	0.6	0.9	1.8	4.1	8.3	17.0
Steps	3	5	10	25	50	100	250	500	1000

(a) 64×64 main model

Time (s)	0.6	0.6	0.9	1.8	3.3	6.3	15.1	30.1	60.4
Steps	3	5	10	25	50	100	250	500	1000

(b) 64×64 main model + $64 \times 64 \rightarrow 256 \times 256$ super-resolution model

Figure 1: Qualitative evaluation of the effect of the number of respacing steps. For each ID vector extracted from the image on the left, we generate images for four seeds with different number of respacing steps, with and without super-resolution. We also report the inference time per image (when using batches of 16 images) in seconds on one NVIDIA RTX 3090 GPU. Note that the time listed for the images with super-resolution also includes the time to generate the 64×64 images.

2. Additional comparisons to state-of-the-art methods

2.1. Qualitative results

Figure 2 shows additional results of the qualitative comparison with the state-of-the-art black-box methods, whose code is available, and demonstrates the superiority of our method both in terms of image quality and identity preservation.

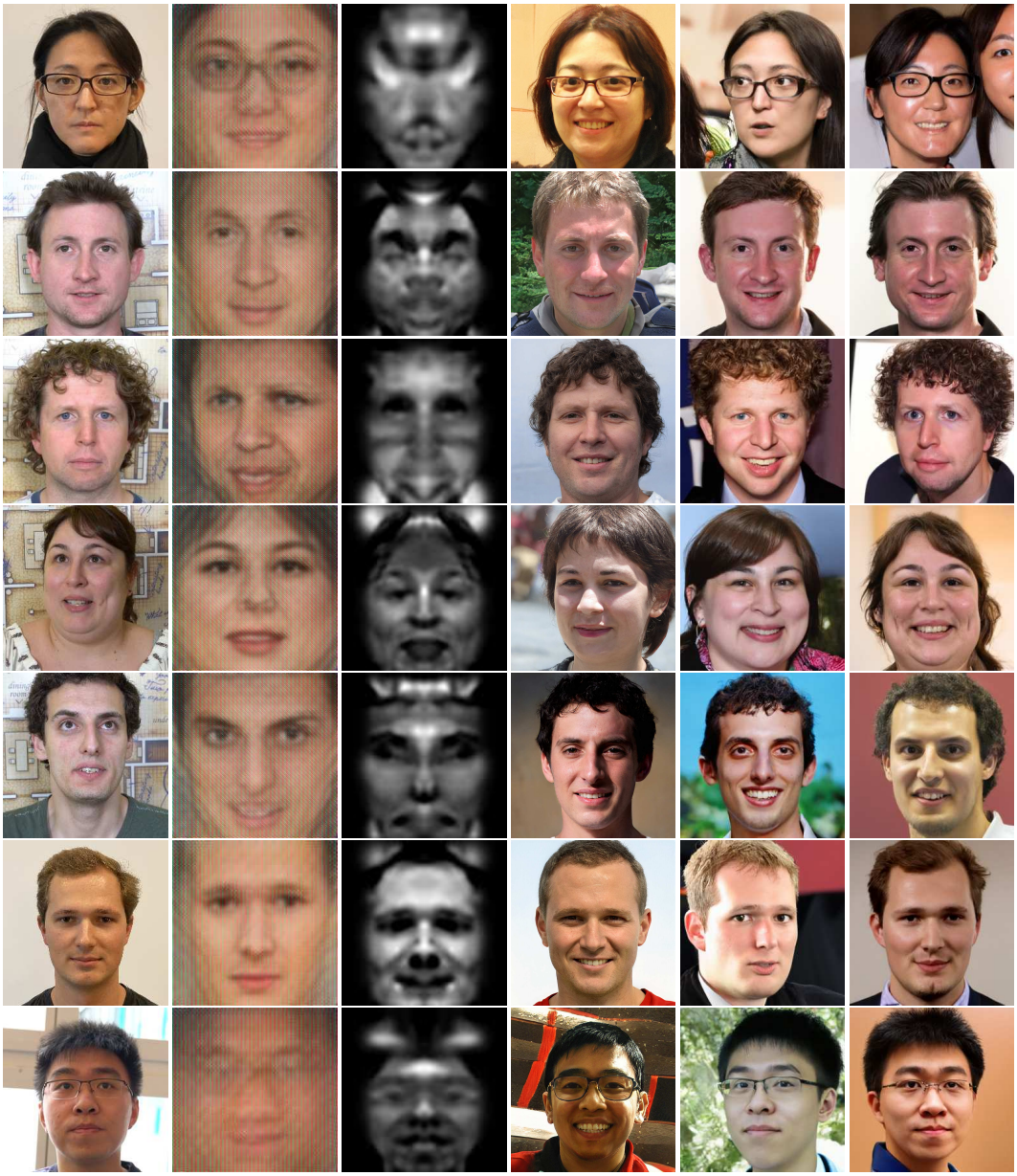


Figure 2: Qualitative evaluation with state-of-the-art methods (additional results). The generated images of our method look realistic and resemble the identity of the original image more closely than any of the other methods.

In the main paper, we quantitatively compare the diversity of our method with that of StyleGAN search [35], which is the only competing method that can produce realistic images at a high resolution. Figure 3 qualitatively confirms that our method produces similarly diverse images but with better identity preservation. For fairness reasons, we use our model trained with FaceNet [28, 8] ID vectors since StyleGAN search [35] uses the same FaceNet implementation [8]. For the first two identities, the StyleGAN search [35] algorithm finds images that share facial features with the original face; however, the identity does not resemble the original face very closely. For the third identity, the search strategy often fails completely by landing in local minima.

Original image StyleGAN search [35] ID3PM (Ours, FaceNet [28, 8])

Figure 3: Qualitative evaluation with StyleGAN search [35]. The generated images of our method resemble the identity of the original image closely and more consistently. Note that StyleGAN search [35] often fails completely for the third identity, whereas our method trained with the same face recognition method (FaceNet [28, 8]) reproduces the identity well.

2.2. User study

To accompany the qualitative results and since we cannot show images from public data sets without the individuals' written consents (as explained in the ethics section of the main paper), we performed a user study with two parts. For the first part, we took the first 10 positive pairs with unique identities according to the LFW [12] protocol (same protocol as used for the quantitative evaluation of the identity preservation in the main paper) to compare the following methods: NbNet [19], Gaussian sampling [24], StyleGAN search [35], ours with FaceNet [28, 8], and ours with InsightFace [3].

Specifically, we instructed the users to:

- Rank the generated images from most **similar to the person of the input image** to least.
- Rank the generated images from most **realistic** to least.

The results in Table 3 (left) provide further evidence of our method outperforming competitor methods, with the exception of StyleGAN search, which achieves better average realism at the expense of identity preservation. The user study also supports our realism labels in the related work section of the main paper.

Method	LFW	
	ID ↓	Real ↓
NbNet [19]	3.52	4.06
Gaussian sampling [24]	4.83	4.90
StyleGAN search [35]	2.53	**1.39**
ID3PM (Ours, FaceNet [28, 8])	**1.90**	2.05
ID3PM (Ours, InsightFace [3])	2.22	2.61

Method	Vec2Face images	
	ID ↓	Real ↓
Vec2Face [7]	3.51	3.80
ID3PM (Ours, FaceNet [28, 8])	2.665	**1.52**
ID3PM (Ours, InsightFace [3])	3.50	3.23
ID3PM (Ours, FaceNet [28, 8], CASIA-WebFace [36])	2.87	3.10
ID3PM (Ours, InsightFace [3], CASIA-WebFace [36])	**2.46**	3.36

Table 3: User study. The listed scores are the mean ranks (1 - 5) for realism (Real) and identity preservation (ID) of the different methods on LFW [12] images (left) and Vec2Face [7] images (right).

For the second part, we took screenshots of the 14 input and result images from Fig. 4 of the Vec2Face [7] paper to compare our method to Vec2Face despite their code not being available. We then computed ID vectors for these faces and generated one image per ID vector with each variation of our method with a fixed random seed and ran a similar user study as above, again with 25 users. We also trained versions of our method on CASIA-WebFace [36][2] to have the same training data as Vec2Face. The results in Table 3 show that all variations of our method beat Vec2Face despite the experimental setup favoring Vec2Face (*e.g.* low-quality screenshots as input for our method and using Vec2Face authors' chosen examples).

2.3. Fairness of comparisons

Our comparisons are fair or to the benefit of competing methods, and no retraining of competing methods was necessary. Gaussian sampling [24] does not have a training data set. StyleGAN search [35] uses a StyleGAN2 [14] trained on all 70000 images[3] of FFHQ [13], so it saw the 1k test images used in the main paper during training (unlike our method that was trained only on the first 69000 images). NbNet [19] and Vec2Face [7] will likely not work well when trained only on FFHQ. NbNet reports significantly worse results in their Tab. 4 and section 4.2.1 when not augmenting their data set (which is already orders of magnitudes larger than FFHQ) with millions of images. Vec2Face uses CASIA-WebFace [36], which is $\sim$ 7 times bigger than FFHQ, and needs class information during training. One can thus consider Vec2Face as white-box with a slightly worse face recognition model (trained with knowledge distillation). When training our method with CASIA-WebFace instead of FFHQ, we obtain similar results and also match or outperform Vec2Face as seen in the above user study. Also, for fairness, we used Vec2Face's protocol for the quantitative comparison of the identity preservation. Lastly, note that no images visualized in the paper were included in any method's training data.

[2]Not upscaled from 64×64 to 256×256 for time reasons.

[3]See https://tinyurl.com/ffhq70k.

3. Controllability

To the best of our knowledge, our method is the first black-box face recognition model inversion method that offers intuitive control over the generation process. The mechanisms described in the following section enable the generation of data sets with control over variation and diversity of the identities as well as their attributes.

3.1. Guidance scale

As described in the main paper, the guidance scale of the classifier-free guidance controls the trade-off between the fidelity in terms of identity preservation (higher guidance) and the diversity of the generated faces (lower guidance). Figure 4 shows examples of the generated images for different guidance scales s ranging from 1.0 to 5.0. To improve the performance for high guidance scales, we adopt dynamic thresholding from Imagen [26] with a threshold of 0.99.

(a) FaceNet [28, 8]

(b) InsightFace [3]

Figure 4: Qualitative evaluation of the effect of the guidance scale. For each ID vector extracted from the image on the left, we generate images for four seeds at guidance scales s ranging from 1.0 to 5.0.

In the main paper, we evaluate the diversity in terms of the pairwise pose, expression, and LPIPS [38] feature distances among generated images as well as their identity embedding distances. To further quantify the effect of the guidance, we select the first 10000 images of the FFHQ data set [13], extract their ID vectors, and generate one image for each ID vector[4]. These 10000 images are then compared to the corresponding 10000 original images in terms of their FID scores [11] as well as precision and recall [17]. The results are shown in Tab. 4. The precision score assesses to which extent the generated samples fall into the distribution of the the real images. Guidance scales in the range $s = 1.5$ to $s = 2.0$ raise the precision score, implying a higher image quality of the generated images. Even larger guidance scales lead to lower precision scores, which could be explained by the saturated colors observed in Fig. 4. The recall score measures how much of the original distribution is covered by the generated samples and corresponds to the diversity. As the guidance scale increases, recall decreases. Similarly, the FID score gets worse with higher guidance scales, demonstrating the decrease in diversity among the generated images.

Method ID vector	Guidance scale s	FID (↓)	Precision (↑)	Recall (↑)
ID3PM (Ours, FaceNet [28, 8])	1.0	**8.014**	0.768	**0.498**
	1.5	9.141	0.782	0.490
	2.0	10.434	**0.783**	0.476
	2.5	11.659	0.775	0.452
	3.0	12.806	0.771	0.441
ID3PM (Ours, InsightFace [3])	1.0	**6.786**	0.774	**0.517**
	1.5	7.442	**0.782**	0.516
	2.0	8.497	0.771	0.508
	2.5	9.286	0.763	0.506
	3.0	10.119	0.746	0.488

Table 4: Quantitative evaluation of the effect of the guidance scale on image quality and diversity. The best performing setting for each ID vector is marked in bold.

Additionally, we perform the face verification experiment from the main paper on LFW [12], AgeDB-30 [20], and CFP-FP [29] with guidance scales between 1.0 and 3.0 for our models trained using FaceNet [28, 8] and InsightFace [3] ID vectors. As seen in Tab. 5, the face verification accuracy generally increases with higher guidance scales but saturates eventually, confirming our qualitative findings that the guidance aids in the identity preservation.

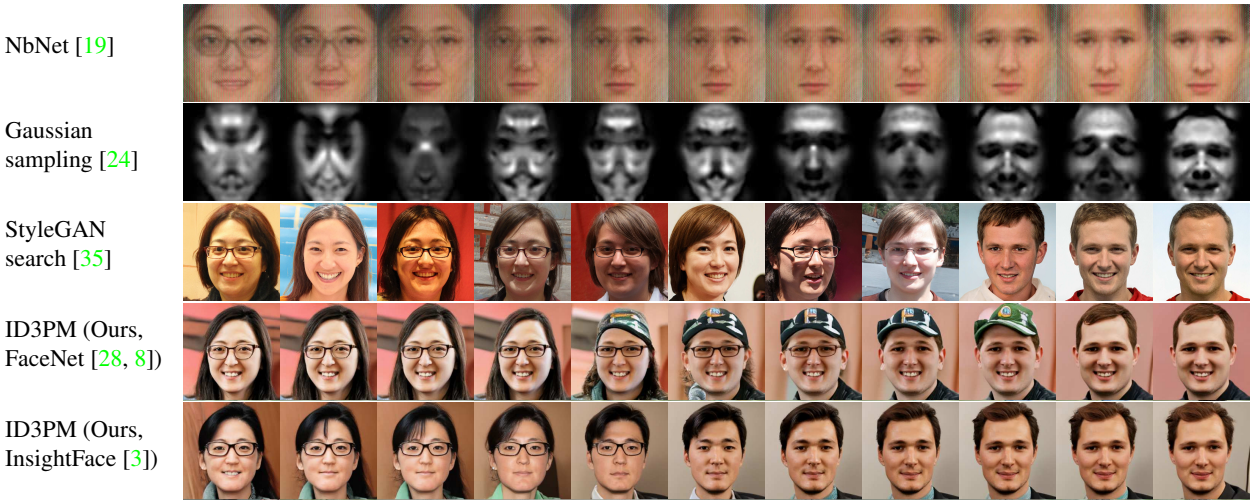
Method	Guidance scale s	LFW		AgeDB-30		CFP-FP	
		ArcFace ↑	FaceNet ↑	ArcFace ↑	FaceNet ↑	ArcFace ↑	FaceNet ↑
Real images	-	99.83%	99.65%	98.23%	91.33%	98.86%	96.43%
ID3PM (Ours, FaceNet [28, 8])	1.0	95.60%	98.62%	84.07%	86.20%	91.83%	94.30%
	1.5	97.08%	99.00%	87.55%	87.90%	94.20%	95.04%
	2.0	97.65%	98.98%	88.22%	88.00%	94.47%	**95.23%**
	2.5	97.92%	98.92%	**88.75%**	**88.47%**	**94.61%**	95.19%
	3.0	**98.03%**	**99.07%**	88.45%	**88.47%**	94.47%	95.03%
ID3PM (Ours, InsightFace [3])	1.0	98.38%	94.37%	91.88%	75.60%	93.50%	85.26%
	1.5	98.95%	95.62%	93.88%	77.57%	95.59%	86.81%
	2.0	**99.20%**	96.02%	94.53%	79.15%	96.13%	87.43%
	2.5	98.97%	**96.37%**	**94.88%**	79.20%	96.03%	87.83%
	3.0	99.15%	96.30%	94.78%	**79.25%**	**96.16%**	**87.97%**

Table 5: Quantitative evaluation similar to the main paper but with different values for the classifier-free guidance s for our models trained using ID vectors from FaceNet [28, 8] and InsightFace [3]. The best performing setting for each ID vector is marked in bold.

[4]Note that we use the generated images of size 64×64 (rather than the upsampled images) for computational reasons.

3.2. Identity vector latent space

Our method is the first to our knowledge to enable smooth interpolations in the ID vector latent space. While we can condition other methods on an interpolated or adapted ID vector as well, their results lack realism and/or do not transition smoothly between images. This is demonstrated in the identity interpolations in Fig. 5. Note that spherical linear interpolation was used for all methods, but linear interpolation leads to a similar performance. The other one-to-many approaches, Gaussian sampling [24] and StyleGAN search [35], were extended such that the seed of all random number generators is set before each image is generated to eliminate discontinuities due to the randomness of the generation process. Nevertheless, certain identity-agnostic characteristics, such as the the expression, pose, and background for StyleGAN search [35], change from one image to the next.



(a) Identity 1 ⟷ Identity 2

(b) Identity 3 ⟷ Identity 4

Figure 5: Identity interpolations for two pairs of identities using state-of-the-art methods. Our method is the only method that provides realistic, smooth interpolations.

As described in the main paper, we can find custom directions in the ID vector latent space. This allows us to change certain identity-specific features such as the age, glasses, beard, gender, and baldness during image generation by traversing along a given direction as visualized in Fig. 6.

(a) Identity 1 (b) Identity 2

Figure 6: Controllable image generation through custom directions in the (InsightFace [3]) ID vector latent space.

3.3. Attribute conditioning

To help disentangle identity-specific from identity-agnostic features as well as to obtain additional intuitive control, we propose attribute conditioning in the main paper. We consider three sets of attributes from the FFHQ metadata [1][5] as shown in Tab. 6 by grouping them by how much they contribute to a person's identity. For example, set 1 only contains attributes that are identity-agnostic whereas set 3 also contains attributes that are strongly correlated with identity. In practice, we recommend using set 1 (and thus use that in the main paper) but show sets 2 and 3 for completeness. Note that the attributes from the FFHQ metadata [1] are in the form of JSON files. Since the neural networks used to extract the attributes are not publicly available, only black-box approaches such as ours that do not require the neural networks' gradients can be used. The attribute conditioning is thus an example of how our proposed conditioning mechanism can be extended to include information from non-differentiable sources.

Attribute	Number of categories	Set		
		1	2	3
Age	1	✗	✗	✓
Emotion	8	✓	✓	✓
Facial hair	3	✗	✗	✓
Hair	8	✗	✗	✓
Head pose	3	✓	✓	✓
Gender [1]	1	✗	✗	✓
Glasses	1	✗	✓	✓
Makeup	2	✗	✓	✓
Occlusions	3	✗	✓	✓

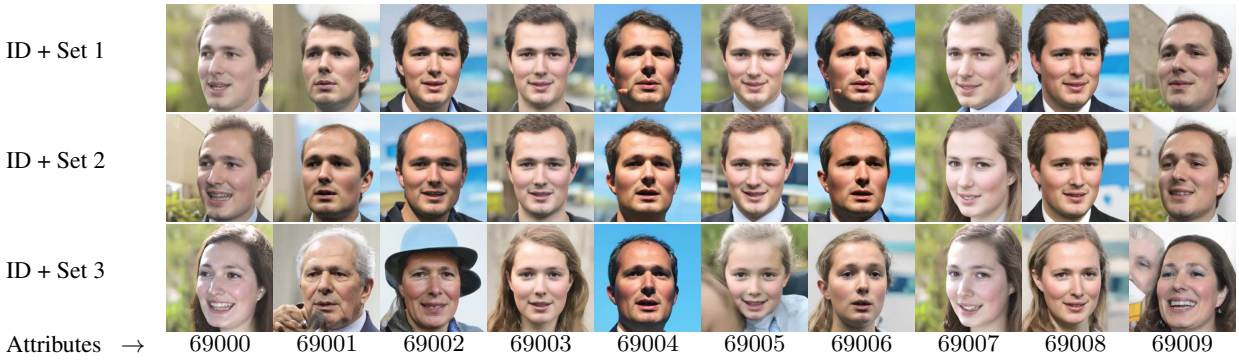
Table 6: Different attribute sets. The number of (potentially identity-correlated) features increases from left to right. [1] While gender arguably falls on a continuous, nonlinear spectrum, we treat it as a binary variable since only this information is available in the data set.

By training our models with attribute conditioning, we can obtain images that recover more of the original data distribution when we sample conditioned on the identity but using the unconditional setting for the attributes (*i.e.* -1-vector for the attribute vector), as shown in the main paper. However, the attributes can overpower the identity information if the attribute set contains attributes that are heavily correlated with the identity. This is visualized in Fig. 7 where we condition our model trained with InsightFace [3] ID vectors on the same ID vector but different attribute vectors, specifically the ones of the first 10 images of the FFHQ test set ($\{69000, 69001, ..., 69009\}$). As we cannot show images from the FFHQ [13] data set as mentioned in the ethics section of the main paper, we instead supply a table with their main attributes. For example, image 69000 is of a happy 27-year old woman with brown/black hair and makeup whose head is turned slightly to the left. For attribute set 1, only the pose and emotion is copied. For attribute set 2, the makeup is also copied. For attribute 3, the gender is also copied, thus leading to an image of a woman for identity 2. In general, as we add more attributes, the original identity is changed increasingly more. Since attribute sets 2 and 3 can alter the identity significantly, we opt for attribute set 1 in most cases.

Interestingly, even when conditioning on attribute set 1 (only pose and emotion), the average identity distance (seen in the quantitative evaluation of the diversity and identity distances in the main paper) increases despite the visual results appearing similar in terms of identity preservation. We hypothesize that this is because most face recognition vectors (inadvertently) encode the pose and expression (see Sec. 5) and are less robust to extreme poses and expressions. Therefore, for the identity distance, it is better to reconstruct a face with a similar pose and expression as the original image. Nevertheless, we argue for the attribute conditioning (using attribute set 1) for most use cases because it leads to more diverse results and allows for an intuitive control over attributes.

[5] We ignore the following attributes from the metadata: smile because it correlates with emotion; and blur, exposure, and noise because we are not interested in them for the purposes of this experiment.

(a) Identity 1



(b) Identity 2

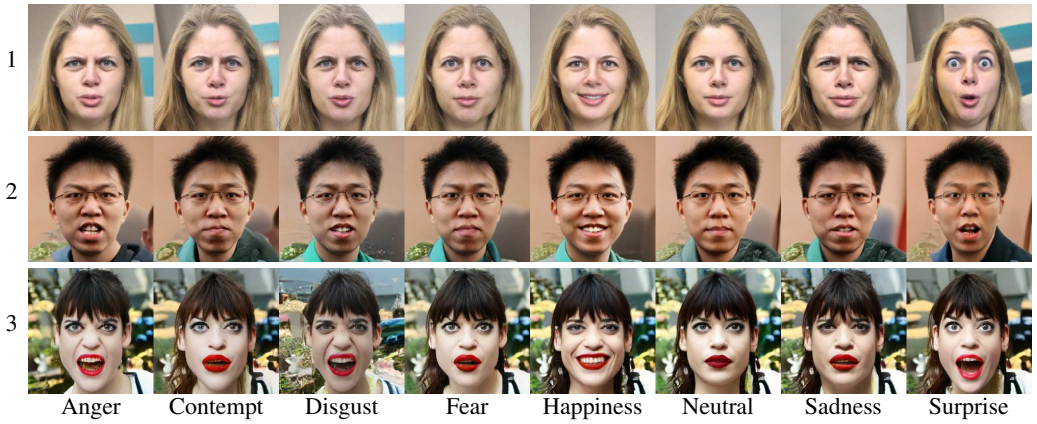
Image number	Gender	Age	Hair	Emotion	Yaw angle	Other
69000	Female	27	Brown/black	Happy	−21.9	Makeup
69001	Male	68	Gray	Neutral	−13.9	-
69002	Female	50	Not visible	Happy	5.5	Headwear + glasses
69003	Female	20	Brown/blond	Happy	−0.4	-
69004	Male	36	Black	Neutral	5.3	-
69005	Female	7	Blond	Happy	−2.6	-
69006	Female	18	Blond	Neutral	9.0	-
69007	Female	21	Brown	Happy	−24.3	Makeup
69008	Female	28	Blond	Happy	10.3	Makeup
69009	Female	43	Black/brown	Happy	−16.6	Makeup + glasses

(c) Attribute descriptions

Figure 7: Attribute conditioning for two identities using different attribute sets. Images in each group have the same (InsightFace [3]) ID vector, but the attributes are chosen from images $\{69000, 69001, ..., 69009\}$ of the FFHQ data set.

Through attribute conditioning, we can simply set the values of desired attributes, such as the emotion and head pose, during inference time to control them, as seen in Fig. 8. Note that our method has no internal structure to enforce 3D consistency. The attribute conditioning alone suffices in generating images that preserve the identity and 3D geometry surprisingly well as we traverse the attribute latent space. This intuitive attribute control paves the way towards using our method to create and augment data sets.

(a) Yaw angle



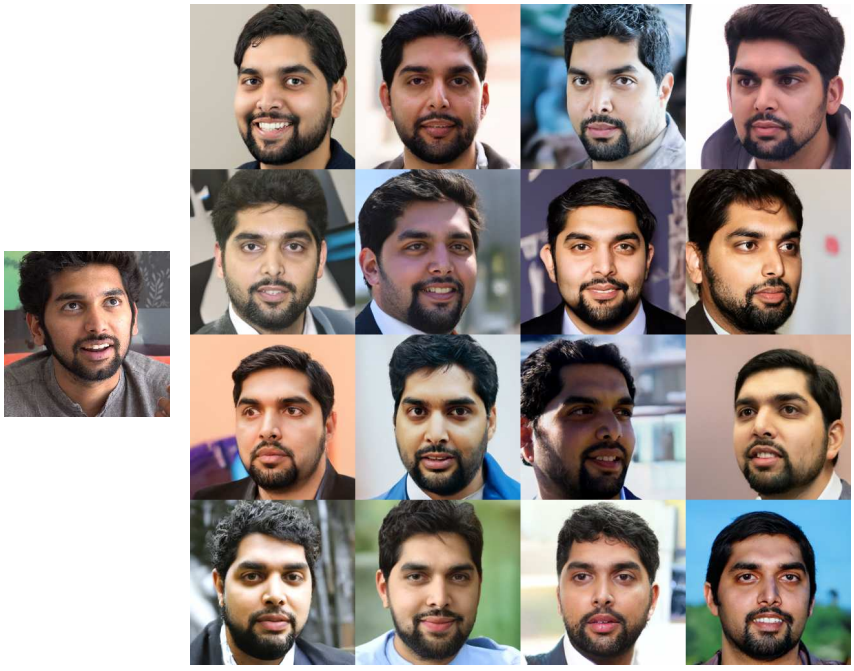
(b) Eight emotions

(c) Sad $\longleftrightarrow$ Happy

Figure 8: Controllable image generation through attribute conditioning. We smoothly change three different attributes of three identities. All models were trained using InsightFace [3] ID vectors and attribute set 1.

4. Failure case

As described in the main paper, one limitation of our approach is that it inherits the biases of both the face recognition model and the data set used to train the diffusion model. This causes our method to sometimes lose identity fidelity of underrepresented groups in the data set, as seen in the example in Figure 9, where the method produces images quite different from the original identity.



Original image ID3PM (Ours, FaceNet [28, 8])

Figure 9: Failure case. Some underrepresented groups in the data sets might have lower identity fidelity for some ID vectors (here FaceNet [28, 8]).

5. Application: Analysis of face recognition methods

With the rise of deep learning and large data sets with millions of images, face recognition methods have reached or even surpassed human-level performance [32, 31, 28, 22]. Nevertheless, face recognition systems have known issues in their robustness to different degradations and attacks [9, 30] and their biases (*e.g.* in terms of ethnic origin) [16, 33, 18].

Due to its general nature with very low requirements for the data set and few assumptions compared to other methods, our method is very well suited for analyzing and visualizing the latent spaces of different face recognition (FR) models. Since our method does not require access to the internals of the pre-trained face recognition model (*black-box* setting), we can analyze different face recognition methods by simply replacing the input ID vectors without worrying about different deep learning frameworks and the memory burden of adding more models.

For the analysis in this section, we train multiple versions of the model with ID vectors extracted with the pre-trained face recognition models listed in Tab. 1. Note that since we specifically want to analyze what identity-agnostic features are contained in common face recognition models, we do not use attribute conditioning here since it would disentangle identity-specific and identity-agnostic features.

5.1. Qualitative evaluation

Figure 10 shows uncurated samples of generated images of the considered ID vectors for several identities. While all generated images appear of a similar quality in terms of realism, the identity preservation of different methods behaves quite differently. The relative performance of different ID vectors changes depending on the identity, but the results for FaceNet [28, 8] and InsightFace [3] seem most consistent on average [6]. As the inversion networks are trained with the same diffusion model architecture and the same data set, we hypothesize that these differences largely boil down to the biases of the respective face recognition methods and the data sets used to train them.

5.2. Robustness

Our method can also be used to analyze and visualize the robustness of face recognition models in difficult scenarios such as varying expressions, poses, lighting, occlusions, and noise as seen in Fig. 11. In line with our previous observations, FaceNet [28, 8] and InsightFace [3] appear the most robust.

In this analysis, it is also relatively easy to tell which features are extracted by observing which features of a target identity's image are preserved. For example, ArcFace [5, 24] and FROM [23] seem to contain pose information as the generated images in the fourth and fifth columns have similar poses as the target identities' images for both identities. Similarly, AdaFace [15] and ArcFace [5, 24] seem to copy the expression for the third column of the first identity. InsightFace [3] also seems to contain expressions and pose for the second identity as seen in columns two to four. Another feature that is commonly copied is whether a person is wearing a hat or not even though this should arguably not be considered part of a person's identity. Interestingly, FROM, a method specifically aimed to mask out corrupted features, does not appear more robust for the tested occlusions (sunglasses, hat). Lastly, noise seems to affect most face recognition methods significantly.

Figure 11 also lists the angular distances of the identities for each generated image to the target identity for the same FR method. The distances for generated images that can be considered failure cases are in general higher than those of the images that worked better. However, they are all still below the optimal threshold[7] calculated for each FR method for real images of the LFW [12] data set using the official protocol – meaning that all generated images are considered to be of the same person. Therefore, we argue that the wrong ID reconstructions are mostly due to problems in the ID vectors rather than the inversion of our model.

We further experiment with out-of-distribution samples such as drawings and digital renders as shown in Fig. 12. Interestingly, despite the extremely difficult setup, some of the resulting images resemble the identity fairly well, especially for FaceNet [28, 8], demonstrating that some face recognition models can extract reasonable identity-specific features even from faces that are out of distribution. Furthermore, this experiment shows that our method can generate extreme features, such as the large nose of the man in the fifth row with FaceNet [28, 8], that are likely not in the training data set.

[6] Note that more images than the representative ones shown here were considered to make this statement and other statements in this section.

[7] It is actually the mean threshold over the 10 different splits. Note that the standard deviation across the splits is smaller than 0.005.

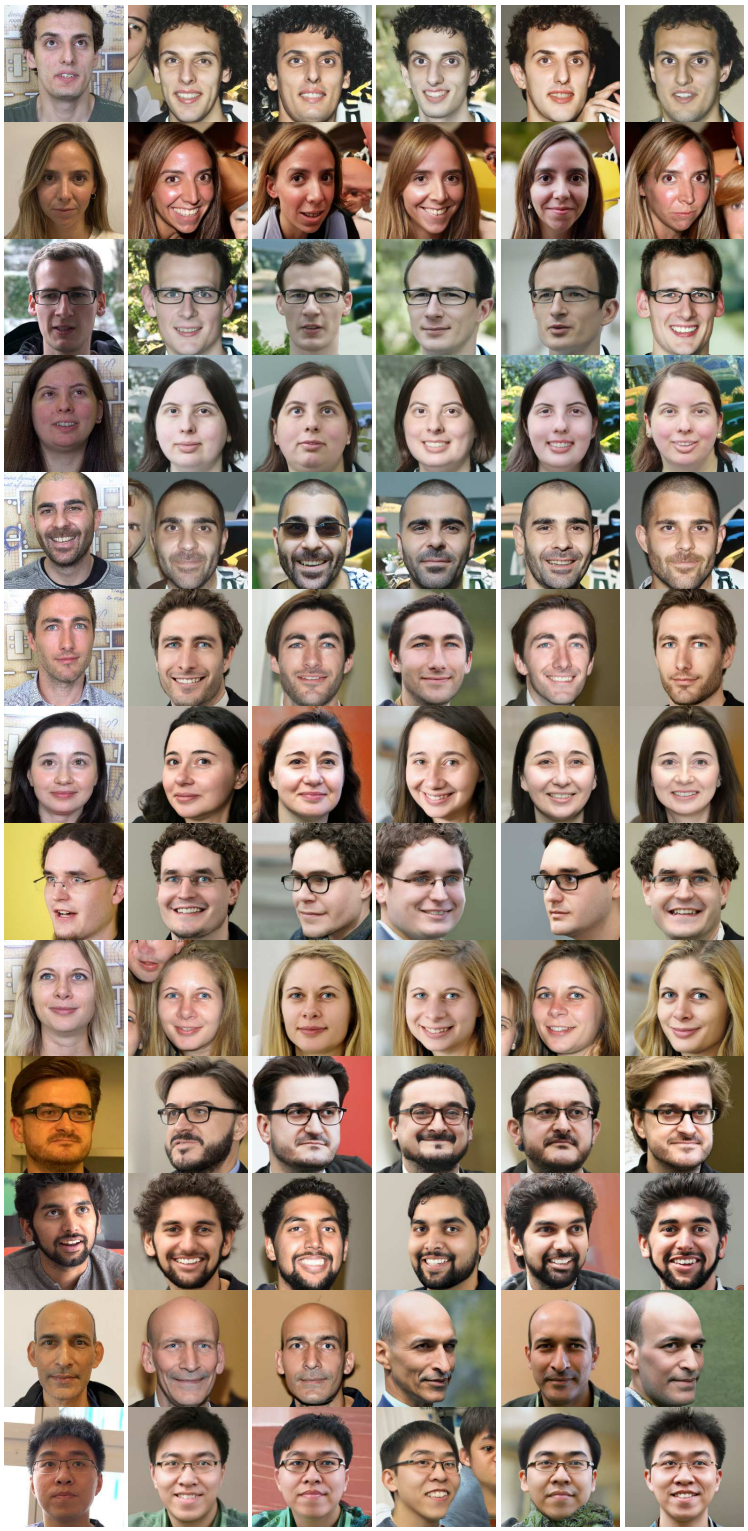
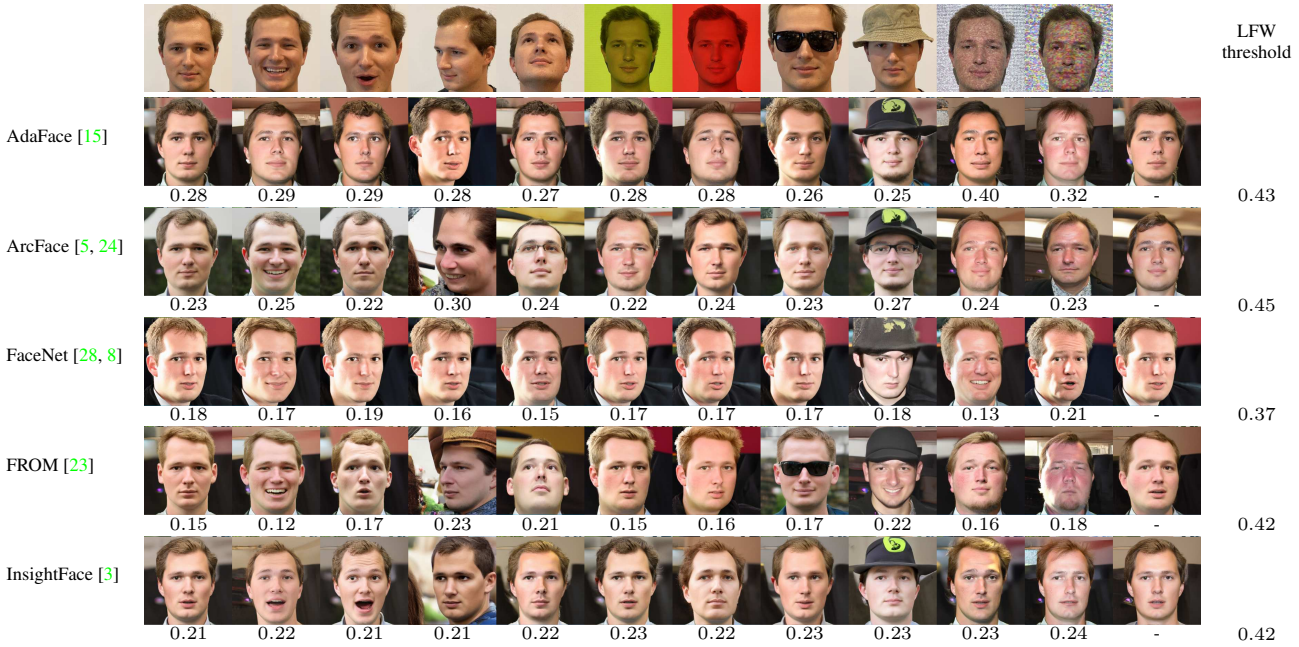


Figure 10: Qualitative evaluation of ID vectors from different state-of-the-art face recognition models. Note that the same seed was used for all images to obtain the most fair results.

(a) Identity 1



(b) Identity 2

Figure 11: Robustness experiment. The first row shows images of a source identity in challenging scenarios whereas the remaining rows show the results when using different ID vectors. The last image column shows images generated from the mean ID vector of all of the source identity images (for which no source image exists). The numbers under each line are the angular distances between the identity of the generated images and the target identity for the same face recognition method. The numbers in the last column are the optimal thresholds for that method calculated using real images of the LFW [12] data set and official protocol.

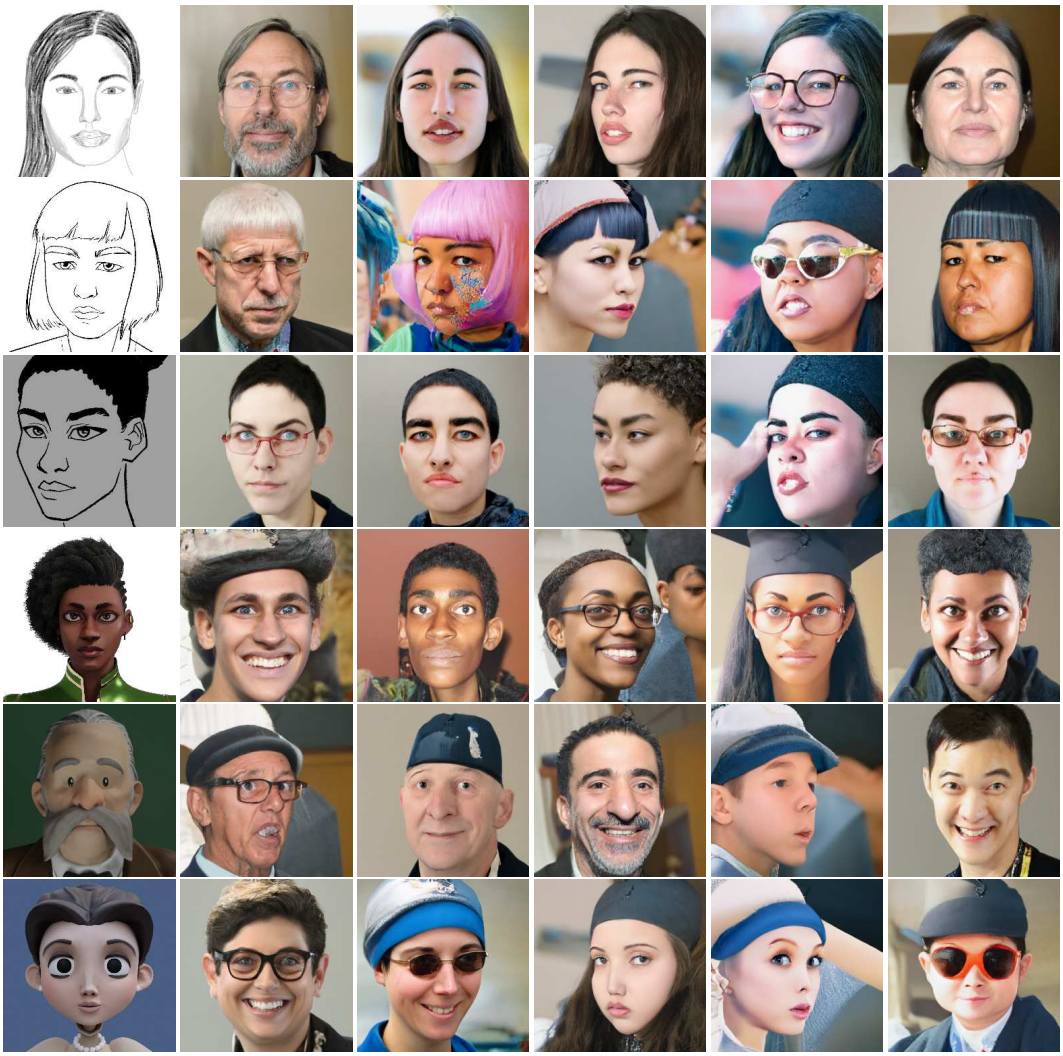
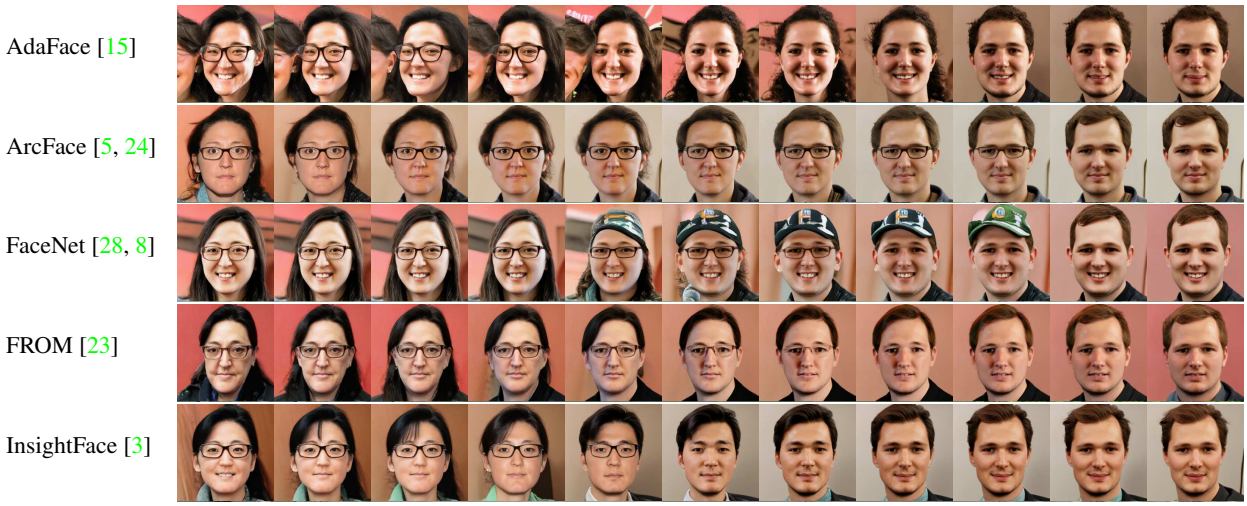


Figure 12: Robustness of ID vectors from different state-of-the-art face recognition models for out-of-distribution samples. Note that the same seed was used for all images to obtain the most fair results.

5.3. Identity interpolations

By interpolating between two ID vectors, our method can produce new, intermediate identities, as shown in the main paper and in Fig. 13. This empirically demonstrates that the latent spaces of most face recognition methods are fairly well-structured. Note that we use spherical linear interpolation because it produces slightly smoother results compared to linear interpolation.



(a) Identity 1 ⟷ Identity 2

(b) Identity 3 ⟷ Identity 4

Figure 13: Identity interpolations for two pairs of identities using different ID vectors.

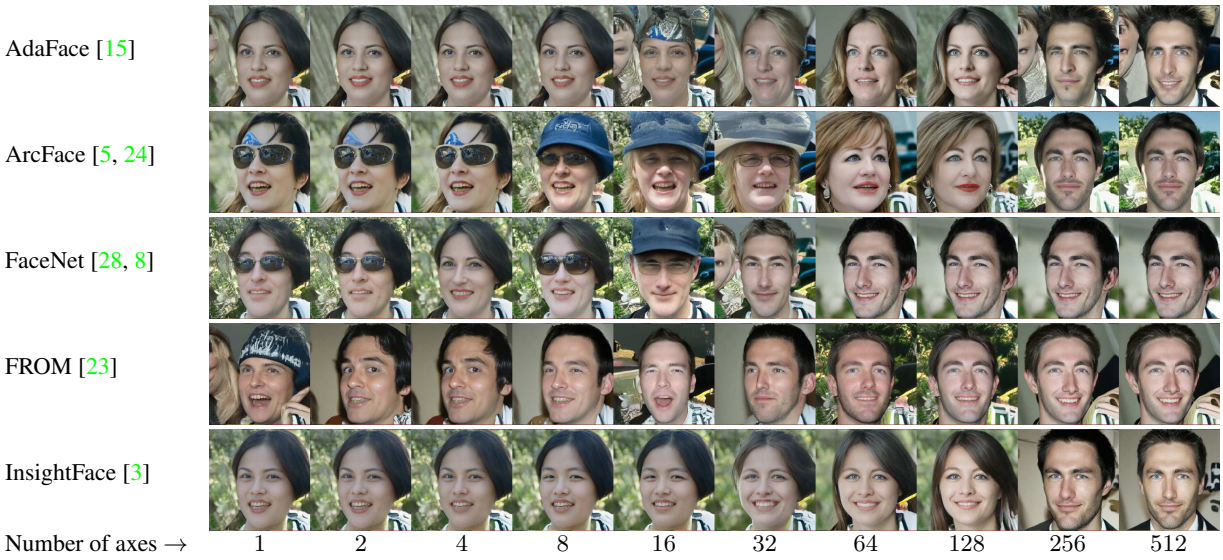
5.4. Principal component analysis

To analyze the most prominent axes within the latent space, we perform principal component analysis (PCA) on the ID vectors of all 70000 images of the FFHQ data set for all considered face recognition models. As seen in Fig. 14, the first principal component appears to mainly encode a person's age while the subsequent components are more entangled and thus less interpretable. Note that we normalize the size of the steps along the PCA directions by the L_2 norm of the ID vectors to ensure a similar relative step size for the different ID vectors. Further note that large steps along any direction can cause the resulting latent vector to leave the distribution of plausible ID vectors and can cause artifacts, which is expected.

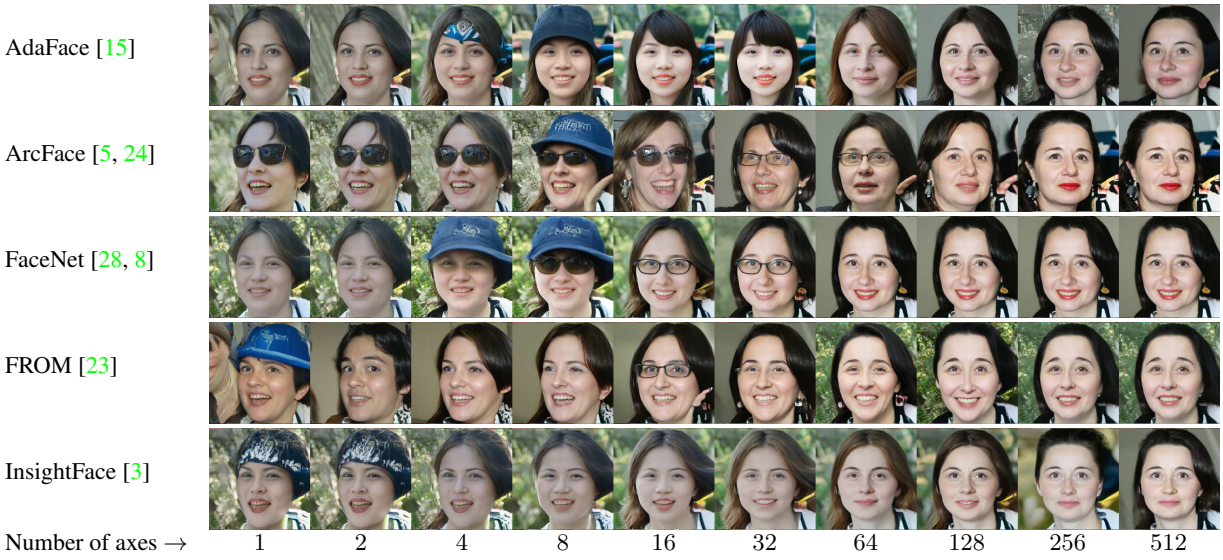
(a) Identity 1 (b) Identity 2

Figure 14: Visualization of the first three principal component analysis axes for two identities using different ID vectors.

Rather than traversing along PCA directions, Fig. 15 shows how the images change when projecting the ID vectors onto the first $\{1, 2, 4, 8, 16, 32, 64, 128, 256, 512\}$ PCA axes. The main insight from this experiment is that the ID vectors from some face recognition models, such as FaceNet [28, 8], can be compressed to as few as 64 dimensions without changing the perceived identity while others, such as AdaFace [15], require all 512 dimensions.



(a) Identity 1



(b) Identity 2

Figure 15: Visualization of the projections onto the first $\{1, 2, 4, 8, 16, 32, 64, 128, 256, 512\}$ principal component analysis (PCA) axes for two identities using different ID vectors.

5.5. Custom directions

Since the PCA axes are difficult to interpret, we calculate custom directions for each face recognition model as described in the main paper. As the biases of the FFHQ data set used to train our inversion models are the same for all ID vectors (*e.g.* glasses appearing when increasing the age direction), the presence or absence of certain directions in the latent space along which a given feature can be changed give insights about what information is extracted by a given face recognition model. As seen in the main paper and Fig. 16, directions that are expected to be contained in the ID vector, such as the age and the gender, can be traversed smoothly. Furthermore, directions that may or may not be considered as part of the identity, such as the current look of a person (*e.g.* glasses, hair style, facial hair style), are also commonly contained as seen in the examples with the blond hair color in Fig. 16.

(a) Identity 1 | Age

(b) Identity 1 | Blond hair color

(c) Identity 2 | Age

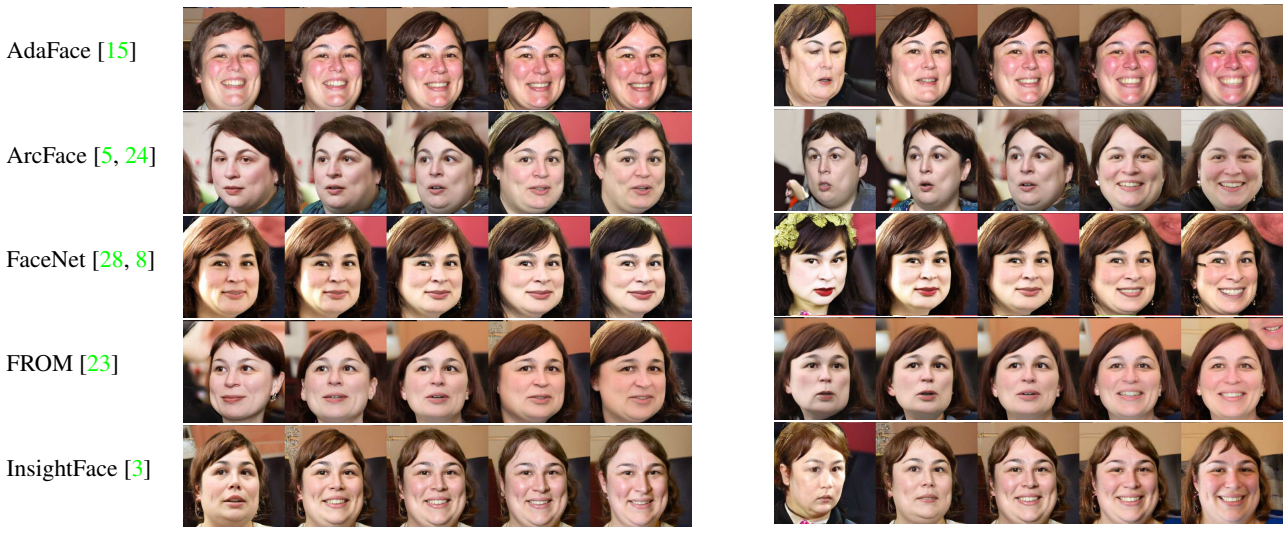
(d) Identity 2 | Blond hair color

Figure 16: Visualization of custom direction modifications for two identities using different ID vectors for two directions that can be considered to belong to a person's identity.

Most interestingly, our method reveals that some directions, such as the pose and emotion of a person, that arguably do not belong to a person's identity can be found for some face recognition models as seen in Fig. 17. For example, ArcFace [5, 24], FROM [23], and InsightFace [3] seem to (inadvertently) extract pose information as the yaw angle can be controlled somewhat by moving along the corresponding direction in the ID vector latent space. Similarly, the smile appears to be controllable in some small region for all considered ID vectors. Note that the goal of looking at these identity-agnostic directions in the ID vector latent space is not necessarily to control this specific dimension cleanly (this can be achieved with attribute conditioning), but rather to analyze what information is extracted by a given FR method. Thus, our method can be used as a tool to reveal and visualize problems of FR methods that we might not even have been aware of and thus suggest hypotheses for further quantitative experiments.

(a) Identity 1 | Yaw angle

(b) Identity 1 | Smile



(c) Identity 2 | Yaw angle

(d) Identity 2 | Smile

Figure 17: Visualization of custom direction modifications for two identities using different ID vectors for two directions that arguably do not belong to a person's identity. Our method reveals that many face recognition methods inadvertently extract identity-agnostic information such as the pose and emotion.

References

[1] DCGM. Gender, age, and emotions extracted for flickr-faces-hq dataset (ffhq), 2020. 11

[2] Jinakang Deng, Jia Guo, Xiang An, Jack Yu, and Baris Gecer. Distributed arcface training in pytorch, 2021. 1

[3] Jinakang Deng, Jia Guo, Xiang An, Jack Yu, and Baris Gecer. Insightface: 2d and 3d face analysis project, 2022. 1, 4, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19, 20, 21, 22, 23

[4] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5203–5212, 2020. 1

[5] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 1, 15, 16, 17, 18, 19, 20, 21, 22, 23

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 1

[7] Chi Nhan Duong, Thanh-Dat Truong, Khoa Luu, Kha Gia Quach, Hung Bui, and Kaushik Roy. Vec2face: Unveil human faces from their blackbox features in face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6132–6141, 2020. 1, 6

[8] Tim Esler. Face recognition using pytorch, 2021. 1, 4, 5, 6, 7, 8, 9, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[9] Gaurav Goswami, Nalini Ratha, Akshay Agarwal, Richa Singh, and Mayank Vatsa. Unravelling robustness of deep learning based face recognition against adversarial attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 15

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 8

[12] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. 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. 6, 8, 15, 17

[13] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 6, 8, 11

[14] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020. 6

[15] Minchul Kim, Anil K Jain, and Xiaoming Liu. Adaface: Quality adaptive margin for face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18750–18759, 2022. 1, 15, 16, 17, 18, 19, 20, 21, 22, 23

[16] Adam Kortylewski, Bernhard Egger, Andreas Schneider, Thomas Gerig, Andreas Morel-Forster, and Thomas Vetter. Empirically analyzing the effect of dataset biases on deep face recognition systems. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 2093–2102, 2018. 15

[17] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32, 2019. 8

[18] Chang Liu, Xiang Yu, Yi-Hsuan Tsai, Masoud Faraki, Ramin Moslemi, Manmohan Chandraker, and Yun Fu. Learning to learn across diverse data biases in deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4072–4082, 2022. 15

[19] Guangcan Mai, Kai Cao, Pong C Yuen, and Anil K Jain. Face image reconstruction from deep templates. *arXiv preprint arXiv:1703.00832*, 2017. 4, 6, 9

[20] Stylianos Moschoglou, Athanasios Papaioannou, Christos Sagonas, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Agedb: the first manually collected, in-the-wild age database. In *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 51–59, 2017. 8

[21] OpenAI. guided-diffusion, 2021. 1

[22] Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *British Machine Vision Association*, 2015. 15

[23] Haibo Qiu, Dihong Gong, Zhifeng Li, Wei Liu, and Dacheng Tao. End2end occluded face recognition by masking corrupted features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1, 15, 16, 17, 18, 19, 20, 21, 22, 23

[24] Anton Razzhigaev, Klim Kireev, Edgar Kaziakhmedov, Nurislam Tursynbek, and Aleksandr Petiushko. Black-box face recovery from identity features. In *European Conference on Computer Vision*, pages 462–475. Springer, 2020. 1, 4, 6, 9, 15, 16, 17, 18, 19, 20, 21, 22, 23

[25] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 1

[26] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 7

[27] David Sandberg. Face recognition using tensorflow, 2018. 1

[28] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 1, 4, 5, 6, 7, 8, 9, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[29] Soumyadip Sengupta, Jun-Cheng Chen, Carlos Castillo, Vishal M Patel, Rama Chellappa, and David W Jacobs. Frontal to profile face verification in the wild. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE, 2016. 8

[30] Richa Singh, Akshay Agarwal, Maneet Singh, Shruti Nagpal, and Mayank Vatsa. On the robustness of face recognition algorithms against attacks and bias. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13583–13589, 2020. 15

[31] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. *Advances in neural information processing systems*, 27, 2014. 15

[32] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014. 15

[33] Philipp Terhörst, Jan Niklas Kolf, Marco Huber, Florian Kirchbuchner, Naser Damer, Aythami Morales Moreno, Julian Fierrez, and Arjan Kuijper. A comprehensive study on face recognition biases beyond demographics. *IEEE Transactions on Technology and Society*, 3(1):16–30, 2021. 15

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 1

[35] Edward Vendrow and Joshua Vendrow. Realistic face reconstruction from deep embeddings. In *NeurIPS 2021 Workshop Privacy in Machine Learning*, 2021. 1, 4, 5, 6, 9

[36] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014. 6

[37] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10):1499–1503, 2016. 1

[38] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 8