Taming Normalizing Flows: Supplementary Material

In the next sections, we provide additional details, results, and visualizations, further demonstrating our method's applications. Additional videos are available in the supplied webpage "`videos.html`".

A. Additional details

First, we elaborate on technical details regarding the implementation of our method, as explained in Sec. 4. We trained a Glow [6] base model to produce RGB images with dimensions 128×128. The training was done for 590K iterations with a batch size of 32, for a total of 316.3 hours, using 4 12GB Titan Xp GPUs. The model has 4 blocks of 32 flows, each consisting of activation normalization layers, 1×1 LU decomposed convolution and additive coupling. The model is trained using an Adam [5] optimizer with learning rate $5 \cdot 10^{-5}$ and betas $(\beta_1, \beta_2) = (0.9, 0.999)$. Images are quantized to 5 bits and learned using the continuous dequantization process as done in previous work [6,10]. Since the dequantization introduces the addition of random noise proportional to the size of quantization bins, every likelihood estimation we perform in Sec. 4.1 is averaged over 10 estimations using different random noise. When we compare NLL of different models, it is done by randomly sampling 10,000 images. When sampling, we use a temperature parameter $T = 0.5$. For the forgetting process, we use a threshold of $\delta = 4$ and a bound of $\epsilon = 0.15 \cdot \delta$. We use the hyperparameters $\alpha = 0.6$ and $\gamma = 0.6$ in all our experiments, chosen using a grid search. As we trained θ_B on the training set of CelebA [8], we used the validation set of CelebA as the holdout set in this evaluation and all upcoming demonstrations, unless specified otherwise.

The classifier used in Sec. 4.2 was trained on the attributes of CelebA [8], using a ResNet50 [2] backbone and achieving an AUC > 0.99 for every binary attribute in CelebA on a holdout set.

In Tab. 1, each experiment is averaged over 5 experiments with different identities. The nearest neighbors are chosen using the 5 nearest neighbors, selected using the average cosine distance between the ArcFace [1] face embeddings.

The tamed model used for Fig. 3 is a model that was trained to forget 15 images of an identity, similar to the last row in Tab. 1.

In Fig. 4, each line in the graph is computed by using the tamed model along the process, while randomly sampling 512 latent vectors that are passed through the model and then classified.

A.1. Normality assumption

Next, we discuss the normality assumption as explained in Sec. 3.3. We assumed the NLL distribution of the base model on the training data is normal. To support this assumption, Fig. A.1 visually compares the distribution with a normal estimation, along with QQ-plots that further support this claim. We also performed a *Kolmogorov–Smirnov test* [9] to compare the distribution to a Normal one. The test is performed on 2,000 random samples drawn from the remember set $\mathcal{D}_R$. We perform it on both the training data of the model and unseen data from the same dataset. The p-value of these tests is 0.95 on the training set and 0.54 on the unseen data. To strengthen the normality assumption, we also compare the NLL distribution of a Normalizing Flow that was trained on CIFAR-10 [7]. The QQ–plots and moral estimation can be seen in Fig. A.2. We performed the Kolmogorov-Smirnov test in the same setting, receiving p-values of 0.58 and 0.38 for the training and unseen data from CIFAR-10, respectively.

Combining all these results, on different modalities, suggests that the NLL of the training data samples follows a Normal distribution, meaning that our normality assumption is grounded.

It is worth noting that our method can also be applied without the normality assumption, by using empirical quantiles of the sampled NLL. This way, we can perform our process in a parameter-free setting, without any requirement on the underlying distribution. However, in this case, the notion of forgetting is less powerful, as we use an unknown empirical CDF instead of the normal distribution one (Eq. (13)), meaning the supplied example using $\delta = 4$ will give different (empirically estimated) values, along with the Likelihood Quantile (Eq. (14)).

Next, we analyze the threshold error bound presented in Eq. (10). When using the normality assumption, we analyze the probability that the NLL of a given data point, $x \in \mathcal{D}$,

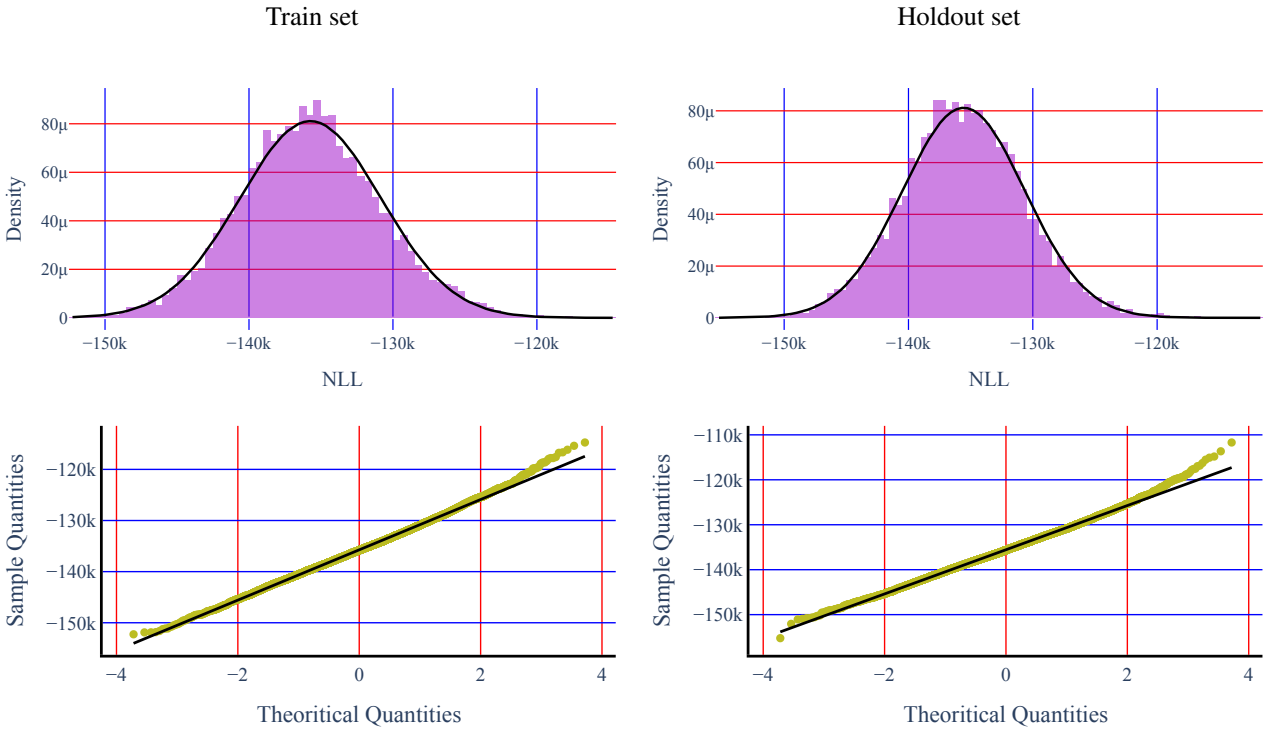

Figure A.1. **Base model (θ_B) NLL normal assumption.** For the base model's training set (left) and a similar holdout set (right), we show a QQ-plot against normal distribution (lower row). We also show (upper row) the normalized density histogram (purple) and a Gaussian estimation (black line) of the distribution. These results suggest that a normal distribution assumption fits this case.

lies inside the error bound:

$$
\begin{gathered}
|\,\mathrm{d}_{\mu_R,\sigma_R}(x,\delta;\theta_T)| < \epsilon \\
\Longleftrightarrow \\
\mu_R + \sigma_R(\delta-\epsilon) < -\log \mathrm{p}_{\theta_T}(x) < \mu_R + \sigma_R(\delta+\epsilon) \\
\Longleftrightarrow \\
\mathrm{NLL}(x) \in [\mu_R + \sigma_R(\delta-\epsilon), \mu_R + \sigma_R(\delta+\epsilon)].
\end{gathered}
$$

Therefore, the probability that the NLL lies insides the error bound is:

$$
\begin{gathered}
P\left(|\,\mathrm{d}_{\mu_R,\sigma_R}(x,\delta;\theta_T)| < \epsilon\right) = \\
\mathrm{F}^{(\mu_R,\sigma_R)}(\mu_R + \sigma_R(\delta+\epsilon)) - \mathrm{F}^{(\mu_R,\sigma_R)}(\mu_R + \sigma_R(\delta-\epsilon)) = \\
\Phi(\delta+\epsilon) - \Phi(\delta-\epsilon).
\end{gathered}
$$

Given our analysis, for $\delta = 4$ and $\epsilon = 0.15\delta$, the probability is roughly 0.033%. This means that when choosing these parameters of δ and ϵ, the probability of an image $x \in \mathcal{D}$ to have a likelihood (in terms of NLL) inside the error bound is low. Specifically in our procedure, when we forget an image $x \in \mathcal{D}_F$ and force its NLL to lie inside the error bound we presented in Eq. (10), its sampling probability is low, meaning the error bound maintains a low sampling probability to the images in $\mathcal{D}_F$.

B. Results

In this section, we discuss additional results associated with experiments from our paper. We show:

(I) Additional scenarios and details for the experiment of "Taming an attribute" (Sec. 4.1).

(II) Results evaluating our method on the remember set $\mathcal{D}_R$.

(III) A full comparison of the experiment of "Taming without the training set" (Sec. 4.3).

(IV) Comparison of the batch size effect on KL loss variance.

(V) Results for an experiment on a dataset from a different modality.

We first discuss the experiment in Sec. 4.1 (Item (I)). Tabs. A.1a to A.1d include a more detailed analysis of Tab. 1, with additional details regarding the likelihood quantiles and the running time. As in Tab. 1, each row in these tables is averaged over 5 different experiments.

For example, the first row in Tab. A.1a shows that when forgetting 1 image, we are able to reach the forget threshold. Regarding the forget set $\mathcal{D}_F$, the likelihood quantile of the

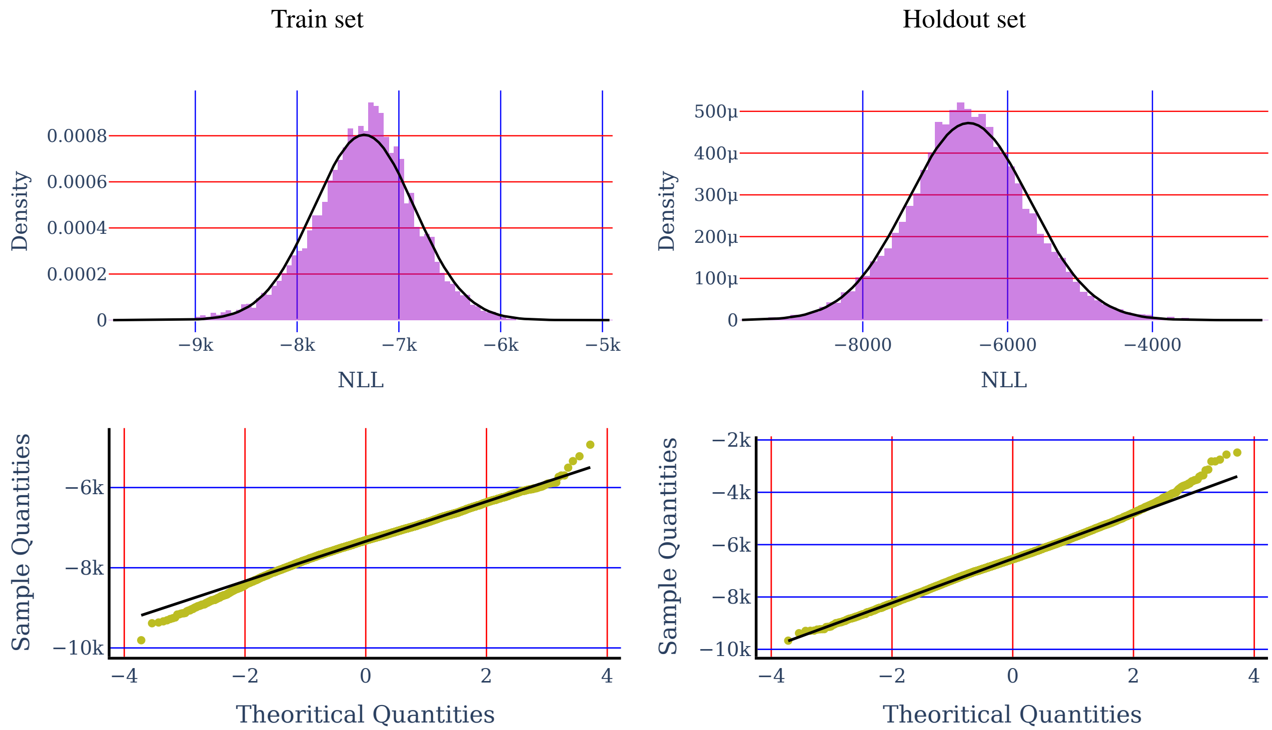

Figure A.2. **Normal assumption on a different modality.** We train a Normalizing Flow on the training set of CIFAR-10 and evaluate it on the training set (left) and test set (right). We show a QQ-plot against normal distribution (lower row). We also show (upper row) the normalized density histogram (purple) and a Gaussian estimation (black line) of the distribution. These results suggest that a normal distribution assumption fits this case as well.

# Images	Forget threshold	Forget set $\mathcal{D}_F$			Forget reference set $\mathcal{D}'_F$			Remember set $\mathcal{D}_R$			Unseen identities $\mathcal{D}_R^{\text{id}}$			Nearest identities $\mathcal{D}_R^{\text{NN}}$			Time[minutes]
		$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	
1	✓	0.47 ± 0.04	$< 10^{-3} \pm 0.00$	0.47 ± 0.04	0.48 ± 0.22	0.47 ± 0.22	$< 10^{-2} \pm 0.03$	0.45 ± 0.00	0.44 ± 0.01	$< 10^{-2} \pm 0.01$	0.36 ± 0.00	0.36 ± 0.01	$< 10^{-3} \pm 0.01$	0.52 ± 0.11	0.51 ± 0.11	0.01 ± 0.01	3.22 ± 1.38
4	✓	0.42 ± 0.17	$< 10^{-4} \pm 0.00$	0.42 ± 0.17	0.48 ± 0.22	0.42 ± 0.23	0.05 ± 0.03	0.45 ± 0.00	0.44 ± 0.00	$< 10^{-2} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.01	0.01 ± 0.01	0.52 ± 0.11	0.50 ± 0.13	0.03 ± 0.02	9.26 ± 1.83
8	✓	0.34 ± 0.13	$< 10^{-4} \pm 0.00$	0.34 ± 0.13	0.48 ± 0.22	0.41 ± 0.25	0.06 ± 0.04	0.45 ± 0.00	0.45 ± 0.00	$< 10^{-3} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-2} \pm 0.00$	0.52 ± 0.11	0.51 ± 0.11	0.01 ± 0.01	16.21 ± 4.01
15	✓	0.38 ± 0.14	$< 10^{-4} \pm 0.00$	0.38 ± 0.14	0.48 ± 0.22	0.36 ± 0.25	0.12 ± 0.04	0.45 ± 0.00	0.45 ± 0.00	$< 10^{-3} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-2} \pm 0.00$	0.52 ± 0.11	0.52 ± 0.11	0.01 ± 0.00	17.60 ± 2.88

(a) Extensive results for the experiment in Tab. 1.

# Images	Forget threshold	Forget set $\mathcal{D}_F$			Forget reference set $\mathcal{D}'_F$			Remember set $\mathcal{D}_R$			Unseen identities $\mathcal{D}_R^{\text{id}}$			Nearest identities $\mathcal{D}_R^{\text{NN}}$			Time[minutes]
		$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	
1	✓	0.47 ± 0.04	$< 10^{-2} \pm 0.00$	0.46 ± 0.04	0.48 ± 0.22	0.47 ± 0.23	0.01 ± 0.01	0.51 ± 0.02	0.51 ± 0.02	$< 10^{-3} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-2} \pm 0.00$	0.52 ± 0.11	0.52 ± 0.11	0.01 ± 0.01	4.16 ± 1.60
4	✓	0.42 ± 0.17	$< 10^{-2} \pm 0.00$	0.42 ± 0.17	0.48 ± 0.22	0.44 ± 0.22	0.03 ± 0.02	0.50 ± 0.03	0.50 ± 0.04	$< 10^{-3} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-2} \pm 0.00$	0.52 ± 0.11	0.51 ± 0.11	0.01 ± 0.01	12.36 ± 3.73
8	✓	0.34 ± 0.13	$< 10^{-2} \pm 0.00$	0.34 ± 0.13	0.48 ± 0.22	0.43 ± 0.23	0.05 ± 0.02	0.49 ± 0.04	0.49 ± 0.04	$< 10^{-2} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-2} \pm 0.00$	0.52 ± 0.11	0.52 ± 0.11	0.01 ± 0.00	19.08 ± 5.32
15	✓	0.38 ± 0.14	$< 10^{-2} \pm 0.00$	0.37 ± 0.14	0.48 ± 0.22	0.39 ± 0.25	0.09 ± 0.04	0.49 ± 0.04	0.49 ± 0.03	$< 10^{-2} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-3} \pm 0.00$	0.52 ± 0.11	0.52 ± 0.11	0.01 ± 0.00	22.29 ± 4.66

(b) Results for a different forget threshold, $\delta = 3$.

# Images	Forget threshold	Forget set $\mathcal{D}_F$			Forget reference set $\mathcal{D}'_F$			Remember set $\mathcal{D}_R$			Unseen identities $\mathcal{D}_R^{\text{id}}$			Nearest identities $\mathcal{D}_R^{\text{NN}}$			Time[minutes]
		$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	
1	✓	0.47 ± 0.04	0.04 ± 0.00	0.43 ± 0.04	0.48 ± 0.22	0.47 ± 0.22	0.01 ± 0.01	0.45 ± 0.03	0.45 ± 0.03	$< 10^{-3} \pm 0.01$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-3} \pm 0.00$	0.52 ± 0.11	0.52 ± 0.11	0.01 ± 0.01	6.09 ± 1.94
4	✓	0.42 ± 0.17	0.03 ± 0.01	0.39 ± 0.16	0.48 ± 0.22	0.46 ± 0.22	0.01 ± 0.01	0.49 ± 0.04	0.49 ± 0.03	$< 10^{-2} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-2} \pm 0.01$	0.52 ± 0.11	0.52 ± 0.11	$< 10^{-2} \pm 0.01$	22.39 ± 13.57
8	✓	0.34 ± 0.13	0.02 ± 0.00	0.32 ± 0.14	0.48 ± 0.22	0.46 ± 0.22	0.02 ± 0.01	0.51 ± 0.03	0.50 ± 0.03	$< 10^{-2} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-2} \pm 0.00$	0.52 ± 0.11	0.52 ± 0.11	0.01 ± 0.00	29.77 ± 8.93
15	✓	0.38 ± 0.14	0.02 ± 0.00	0.35 ± 0.14	0.48 ± 0.22	0.44 ± 0.22	0.04 ± 0.02	0.50 ± 0.02	0.50 ± 0.02	$< 10^{-2} \pm 0.00$	0.36 ± 0.00	0.36 ± 0.00	$< 10^{-2} \pm 0.00$	0.52 ± 0.11	0.52 ± 0.11	0.01 ± 0.00	35.98 ± 10.64

(c) Results for a different forget threshold, $\delta = 2$.

# Images	Forget threshold	Forget set $\mathcal{D}_F$			Forget reference set $\mathcal{D}'_F$			Remember set $\mathcal{D}_R$			Unseen identities $\mathcal{D}_R^{\text{id}}$			Nearest identities $\mathcal{D}_R^{\text{NN}}$			Time[minutes]
		$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	$q_{\theta_B}(\cdot)$	$q_{\theta_T}(\cdot)$	$QD(\cdot)$	
1	✓	0.48 ± 0.05	$< 10^{-3} \pm 0.00$	0.48 ± 0.05	0.32 ± 0.03	0.28 ± 0.03	0.04 ± 0.02	0.50 ± 0.03	0.49 ± 0.03	$< 10^{-2} \pm 0.01$	0.34 ± 0.00	0.34 ± 0.00	$< 10^{-2} \pm 0.00$	0.52 ± 0.14	0.51 ± 0.14	0.01 ± 0.01	3.71 ± 1.07
4	✓	0.40 ± 0.15	$< 10^{-4} \pm 0.00$	0.40 ± 0.15	0.32 ± 0.03	0.24 ± 0.09	0.08 ± 0.08	0.50 ± 0.04	0.50 ± 0.05	$< 10^{-2} \pm 0.00$	0.34 ± 0.00	0.34 ± 0.01	$< 10^{-2} \pm 0.01$	0.52 ± 0.14	0.51 ± 0.14	0.01 ± 0.00	8.69 ± 2.62
8	✓	0.39 ± 0.17	$< 10^{-4} \pm 0.00$	0.39 ± 0.17	0.32 ± 0.03	0.19 ± 0.10	0.14 ± 0.10	0.49 ± 0.01	0.49 ± 0.01	$< 10^{-2} \pm 0.01$	0.34 ± 0.00	0.34 ± 0.00	$< 10^{-2} \pm 0.01$	0.52 ± 0.14	0.51 ± 0.14	0.02 ± 0.02	12.74 ± 3.76
15	✓	0.41 ± 0.14	$< 10^{-4} \pm 0.00$	0.41 ± 0.14	0.32 ± 0.03	0.14 ± 0.10	0.19 ± 0.12	0.51 ± 0.03	0.51 ± 0.03	$< 10^{-2} \pm 0.00$	0.34 ± 0.00	0.34 ± 0.00	$< 10^{-3} \pm 0.00$	0.52 ± 0.14	0.51 ± 0.14	0.02 ± 0.02	21.49 ± 9.76

(d) Results on an identity outside the training set (from a holdout set of the same distribution).

Table A.1. **Forget an identity - Comprehensive evaluation.** Additional results for the experiment in Tab. 1, including results for additional thresholds, and forgetting an identity outside the training set. These tables include the Quantile drop ($QD_{\theta_B,\theta_T}(\cdot)$), along with the likelihood quantiles ($q_\theta(\cdot)$), for different evaluated sets. It also includes the running time in minutes of every experiment.

base model is 0.47 while for the tamed model it is $< 10^{-3}$, resulting in a quantile drop of 0.47. This row also shows that the running time for this experiment is 3.2 minutes.

Tabs. A.1b and A.1c include the results for using a dif-

# Images	Forget threshold	Quantile drop $\mathrm{QD}_{\theta_B,\theta_T}(\cdot)$ (see Eq. (15))				
		$\mathcal{D}_F(\uparrow)$	$\mathcal{D}'_F(\uparrow)$	$\mathcal{D}_R(\downarrow)$	$\mathcal{D}_R^{\mathrm{id}}(\downarrow)$	$\mathcal{D}_R^{\mathrm{NN}}(\downarrow)$
1	✗	$< 10^{-3} \pm 0.01$	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-3} \pm 0.00$	$< 10^{-2} \pm 0.00$
4	✗	$< 10^{-3} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-3} \pm 0.00$
8	✗	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$
15	✗	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$	$< 10^{-2} \pm 0.00$

Table A.2. **Baseline — forget identity.** Similar to Tab. 1, when we compare a naïve approach of forgetting, by resuming to train only on the remember set, we get no forgetting, with minimal change in the distribution. The notations are the same as Tab. 1, with (↑) and (↓) indicating whether higher or lower is better, respectively.

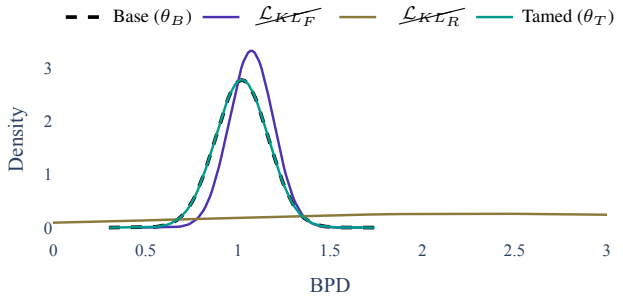

Figure A.3. **Ablation distribution comparison.** Comparison of the NLL distribution of models presented in Sec. 5. Notice how $\mathcal{L}_{KL_R}$ and $\mathcal{L}_{KL_R}$ do not preserve the base distribution well.

ferent forget threshold, $\delta = 2$ and $\delta = 3$ respectively. Tab. A.1d includes results for using a forget set outside of the training set, *i.e.*, $\mathcal{D}_F \not\subset \mathcal{D}_R$. The images in $\mathcal{D}_F$ are all from an identity of a holdout set from the same distribution.

These tables show that even for the aforementioned different settings, we are able to forget the identity ($\mathcal{D}_F$) while reducing the likelihood of a holdout set of its images ($\mathcal{D}'_F$), with marginal impact on the remember distribution ($\mathcal{D}_R$, $\mathcal{D}_R^{id}$ and $\mathcal{D}_R^{NN}$).

To add context to the experiment in Sec. 4.1, we also present results of a baseline experiment. In Tab. A.2 we explore what happens when we do not force anything on the forget images, and we use the original training objective of Normalizing Flows only on the Remember set, *i.e.* $\mathrm{A}_\theta(\mathcal{D}_R)$ (see (Eq. (5)). We observe that neither the forgetting threshold was reached (Eq. (10)) nor any forgetting occurred. As we have two opposite objectives, it is natural to explore a baseline that performs the negative objective on the forget set, *i.e.* $\mathcal{L} = \mathrm{A}_\theta(\mathcal{D}_R) - \mathrm{A}_\theta(\mathcal{D}_F)$. In this case, the distribution diverges, and we receive infinite NLL values quickly, meaning that we do not preserve the structure of the NLL of $\mathcal{D}_R$ at all.

Now we turn to inspect whether the time to forget an identity depends on the number of images the model was trained on. To do so, we trained an additional base model just on CelebA. This model was trained on 162,770 images, while the original one, trained additionally on FFHQ [4], was trained on 232,770. For both models, training stopped with the same performance (in NLL) on CelebA's training set. In Tab. B.1, we compare the running time of these models and see that even for a smaller training set the running time is comparable and fast.

As discussed in Sec. 5, in Fig. A.3 we compare the distribution of NLL values on the training set of the base model, for different ablated models. Some models are not shown in the figure, as they have a distribution that is visually indistinguishable from the shown distributions of the base and tamed models. The figure shows that without using the forward KL divergence loss ($\mathcal{L}_{KL_F}$), the distribution is worse, but it's also more "narrow", fitting the mode-seeking behavior of the reverse KL divergence. On the contrary, without the reverse KL divergence ($\mathcal{L}_{KL_R}$), which is known to be important for generative tasks, the performance is bad, and fits the mean-seeking behavior of forward kl divergence, attempting to cover more regions.

Next, we discuss Item (II), showing how our method preserves the NLL distribution of the remember set $\mathcal{D}_R$. In Sec. 4, we showed results focusing on the forget set $\mathcal{D}_F$. We now show results, focusing on $\mathcal{D}_R$. This is demonstrated by showing this distribution before and after taming, as seen in Fig. A.4. The figure visualizes the differences between the distributions of the base model (θ_B) and the tamed model (θ_T), for both the training set and a holdout set. This is done using the normalized density histogram of these distributions, and also by estimating the parameters of a normal distribution using the distributions' observations. The distribution pairs in Fig. A.4 are all similar, indicating that we successfully forget the target(s), without heavily impacting the rest of the distribution.

Next, we discuss the experiment in Sec. 4.3 (Item (III)). Fig. B.1 shows a more detailed comparison of Fig. 6, additionally showing the NLL distribution of the tamed model (θ_T) on the original training data. We see that while there is some decrease in the likelihood of the original training data, this change is much smaller than the difference be-

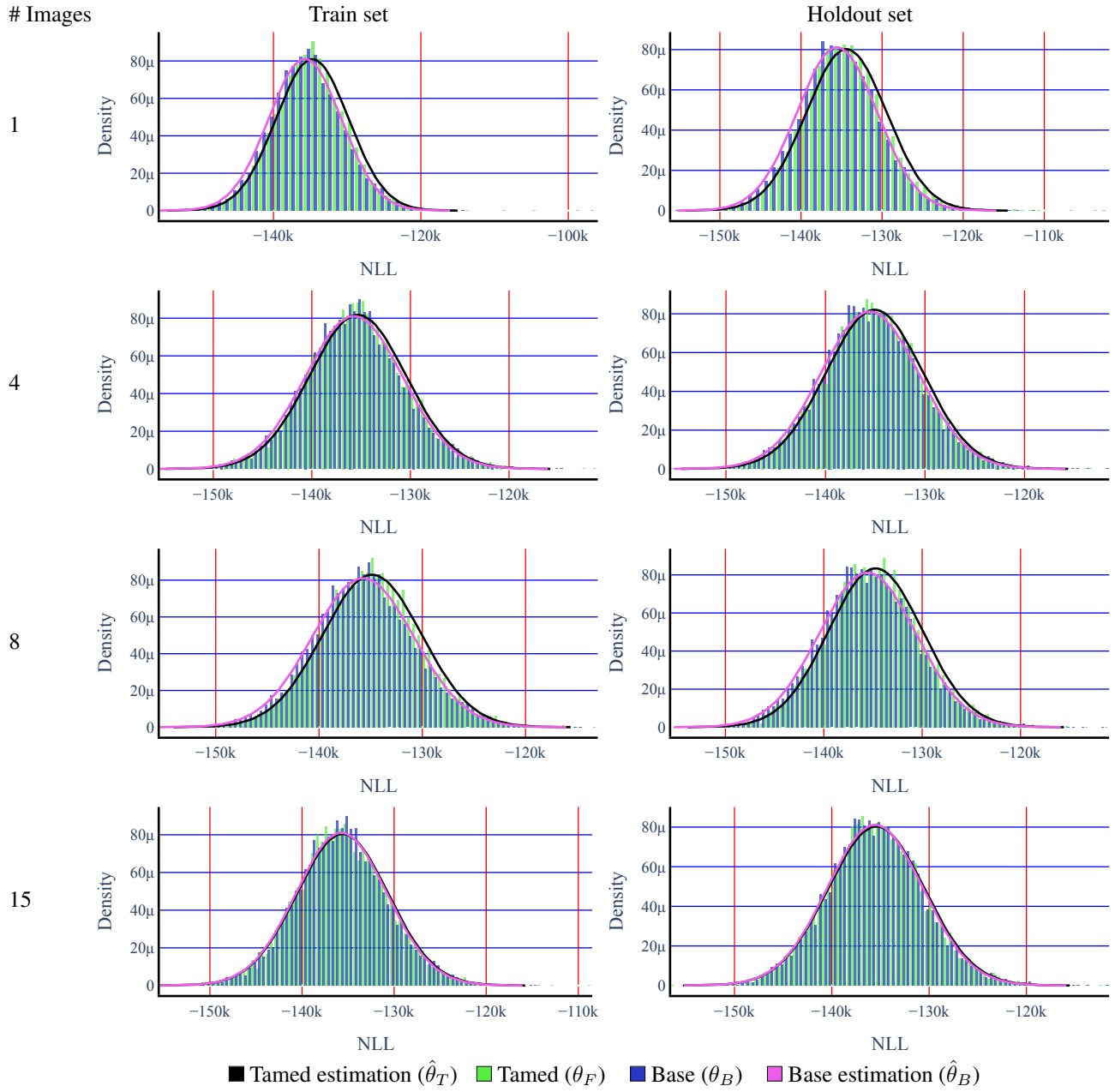

Figure A.4. **Preserving the NLL's distribution of the remember set $\mathcal{D}_R$.** Normalized density histogram and normal estimation of the NLL distribution on the base model training set, and a similar holdout set. The different plots correspond to models that were tamed to forget images of a specific identity, with a varying number of images. These plots suggest that when taming, the change of the $\mathcal{D}_R$ distribution is minor.

tween the original training data $\mathcal{D}$ and the remember set $\mathcal{D}_R$, *i.e.*, $\{-\log \mathrm{p}_{\theta_T}(\mathcal{D})\}$ and $\{-\log \mathrm{p}_{\theta_T}(\mathcal{D}_R)\}$, respectively.

In Fig. B.1, We evaluate the impact of the forget set size ($|\mathcal{D}_F|$) on our method, w.r.t. the experiment in Sec. 4.3. As the figure shows, when the size of $\mathcal{D}_F$ is small (*i.e.*, $|\mathcal{D}_F| < 40$) the average likelihood quantile remains near zero. When $|\mathcal{D}_F| > 40$, the average likelihood quantile increases. This is aligned with the different settings of our method, as we showed in Sec. 4.2 where we used larger sets of forget images $\mathcal{D}_F$.

Next, we discuss the effect of batch size on the variance of the KL loss, $\mathrm{Var}(\mathcal{L}_{KL})$. In every SGD iteration, we compute the distribution of the remember set $\mathcal{D}_R$, according to a sampled batch (see line 3 in Algorithm 1). The size of the sampled batch can affect our loss term. Specifically, we can look at the loss that compares the sampled remem-

# Images	CelebA+FFHQ		CelebA	
	T[minutes]	T[%]	T[minutes]	T[%]
1	3.2	0.02%	3.9	0.04%
4	9.3	0.06%	9.0	0.09%
8	16.2	0.09%	16.5	0.16%
15	17.6	0.15%	21.9	0.21%

Table B.1. **Training size effect on running time.** We compare the running time for taming an identity using two different base models (θ_B), trained using different training set size ($\mathcal{D}$). T[minutes] is the time taken to run this experiment in minutes. T[%] is the experiment's runtime divided by the base model's total training time, in percentages.

# Images	Forget threshold	$\text{QD}_{\theta_B,\theta_T}(\cdot)$ (Eq. (15))	
		$\mathcal{D}_F(\uparrow)$	$\mathcal{D}_R(\downarrow)$
1	✓	0.34 ± 0.36	-0.25 ± 0.01
4	✓	0.24 ± 0.14	-0.24 ± 0.01
8	✓	0.24 ± 0.14	-0.26 ± 0.01
15	✓	0.25 ± 0.15	-0.26 ± 0.01

Table B.2. **Forget — other modality.** When forgetting a different modality (CIFAR-10 [7]) in an experiment similar to Sec. 4.1, we are able to reduce the likelihood of a set of images, in this case images from the same class of objects.

ber distribution and the new distribution, the $\mathcal{L}_{KL}(\cdot)$ loss. As the sample size decreases, the loss's variance increases. Fig. B.2 shows an analysis of this effect. The plots show that when computing the distribution with extremely small batch sizes such as $1, 2, 4$ the variance is extremely high, while it drops for bigger batches.

Lastly, we discuss an experiment on a modality that is different than faces. Tab. B.2 contains results for an experiment on CIFAR-10 [7]. This experiment has a similar setting to Sec. 4.1, forgetting a specific set of images. In this case, instead of focusing on images with the same identity, we focus on images from the same class of objects, *e.g.* airplanes. We see that in this case, with images containing different objects, a domain that is less aligned than human faces, we are also able to forget the specific forget set we intended, without harming the likelihood of other images in the remember set.

C. Visualizations

In this section, we show different generated samples of tamed models from the different experiments in Sec. 4.

We begin with Fig. C.1, showing the generated images when experimenting without any access to the training set as in Sec. 4.3. This figure shows how the similarity between the remember set $\mathcal{D}_R$ and the training set $\mathcal{D}$ affects the generation quality. Using a similar distribution (CelebA validation set) maintains generation quality, while using a more distant distribution (FairFace [3]) does not.

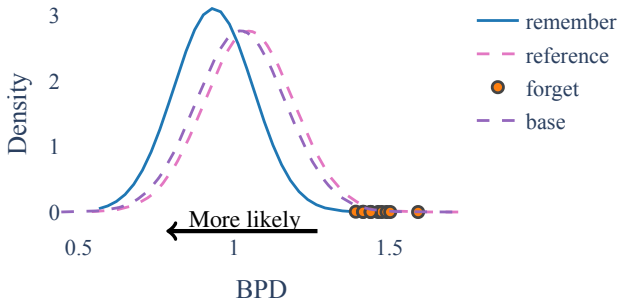

Figure B.1. **Forget without training data access.** Full comparison of Fig. 6. Here we see not only the NLL distributions of θ_T on the remember set (solid), but also on the training set $\mathcal{D}$ (dashed pink). The distribution of the base model θ_B on the training data (dashed purple) is close to the tamed model on the same data, while the forget images (orange dots) reach the forget threshold.

Next, we discuss additional examples for taming an attribute (Sec. 4.2).

In Fig. C.2, we see that an identity possessing blond hair can quickly be scrubbed of that attribute (1^{st} row). Identities without blond hair will obtain a darker hair color as a result of this process, as we globally reduce the blond hair attribute (2^{nd} row). This property can be used to debias a model, *e.g.*, a model that generates images of females with higher probability, can be tamed in order to achieve a higher generation probability of males (as shown in the 7^{th} row). This figure also shows that the changes are related to the data in the forget and remember sets. This can be seen in the 4^{th} row, as the blond hair change on the male identity is less impactful compared to the female ones. This is due to the fact the training data (CelebA) only has 0.85% images of blond males.

Fig. C.3 shows how while we change an attribute globally, when we focus on a single latent vector, even in different experiments, the attribute change is applied while preserving the original identity.

Additional examples that demonstrate different attribute changes (Sec. 4.2) are available in the supplied webpage "`videos.html`". The videos depict the process of our method, including additional examples that describe the effect of coupled attributes in the dataset on our procedure.

References

[1] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF con-*

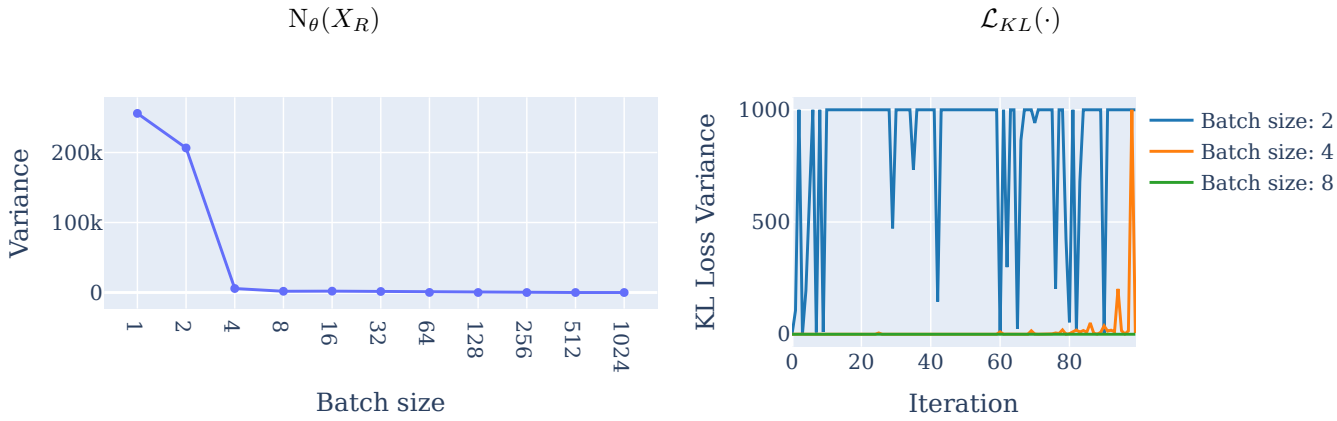

Figure B.2. **Batch size effect on $\mathcal{L}_{KL}$.** Our loss consists of computing a distribution from a sampled batch (see line 3 in Algorithm 1), which is affected by the size of the batch. (**Left**) The variance of the NLL as a function of the batch sizes of samples from $\mathcal{D}_R$. (**Right**) The variance of the KL loss along the iterations of our method. The loss was clipped at 1000 to depict the differences.

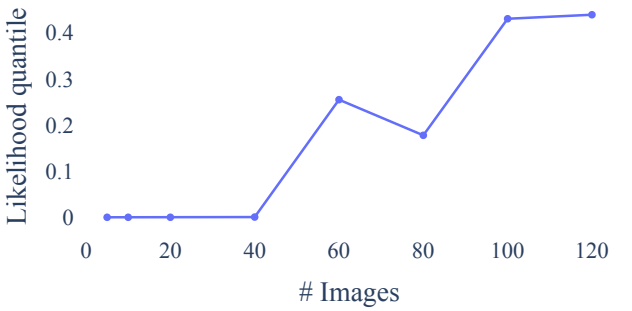

Figure B.3. **Taming success against the number of samples.** The plot shows the likelihood quantile (Eq. (14)) of different models without any access to the training data. $\mathcal{D}_R$ contains 1000 images out of the base model's training set. Each model is tamed to forget a different number of images (the x-axis). We see that when $\mathcal{D}_F$ is small, taming can yield a big likelihood reduction, while on a lot of images, the average difference drops. These were tested on the scenario of Sec. 4.3, on data from FairFace [3].

ference on computer vision and pattern recognition, pages 4690–4699, 2019. 1

[2] 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

[3] Kimmo Kärkkäinen and Jungseock Joo. Fairface: Face attribute dataset for balanced race, gender, and age. *arXiv preprint arXiv:1908.04913*, 2019. 6, 7, 8

[4] 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 (CVPR)*, June 2019. 4

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 1

[6] Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative Flow with Invertible 1x1 Convolutions, July 2018. Number: arXiv:1807.03039 arXiv:1807.03039 [cs, stat]. 1

[7] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 1, 6

[8] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015. 1

[9] F. J. Massey. The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American Statistical Association*, 46(253):68–78, 1951. 1

[10] Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015. 1

Figure C.1. **Taming with no training set.** Images generated using models that were tamed without any training data access, along the number of iterations of our method (0, 20, 40). The left side shows that as the remember set is more distant than the training set (Fairface [3]), the results are worse compared to unseen data from a closer distribution (CelebA validation set).

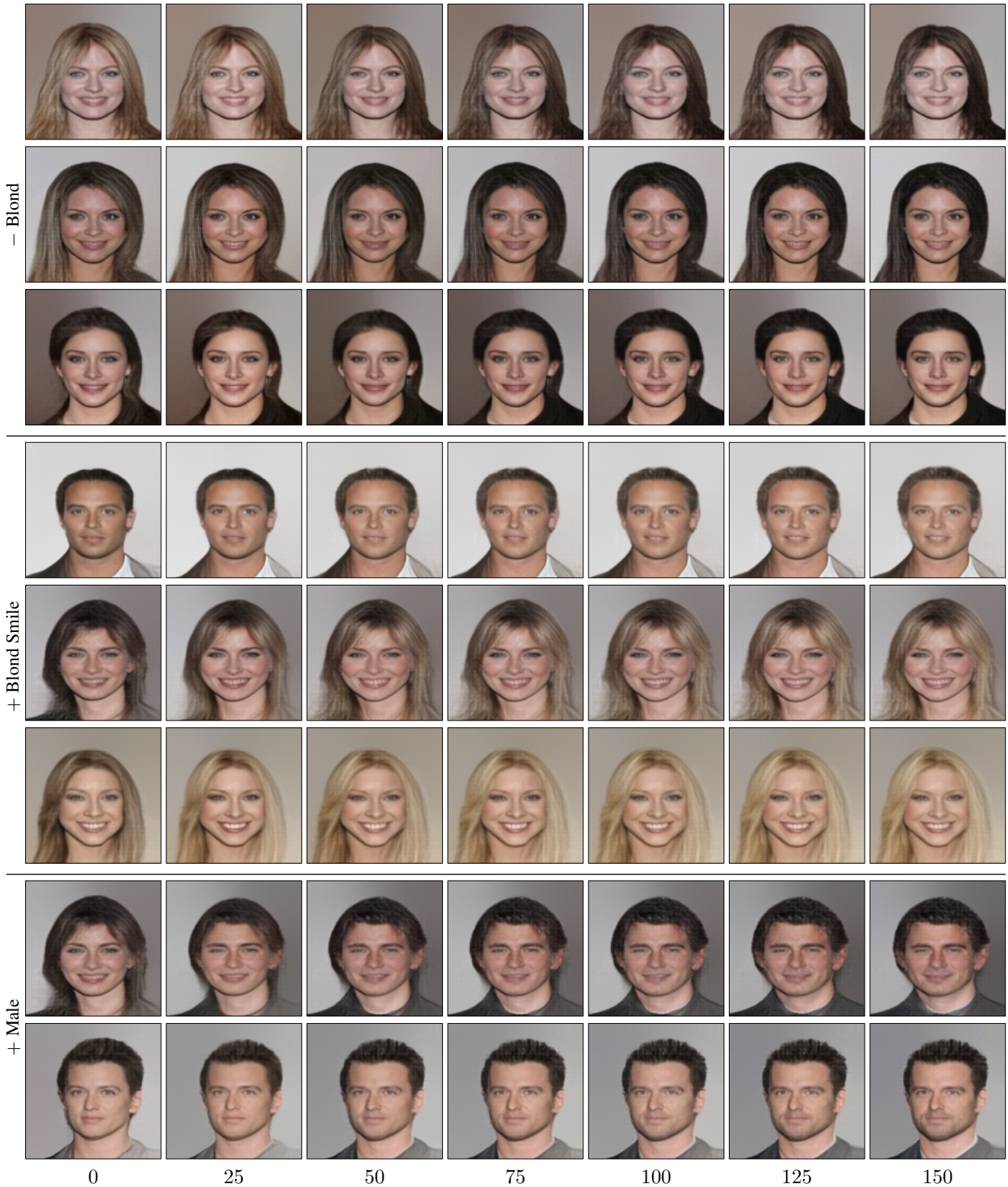

Figure C.2. **Change attributes process.** By examining the same latent vectors during our process, we are able to visualize the change of different attribute (the bottom row includes the number of iterations).

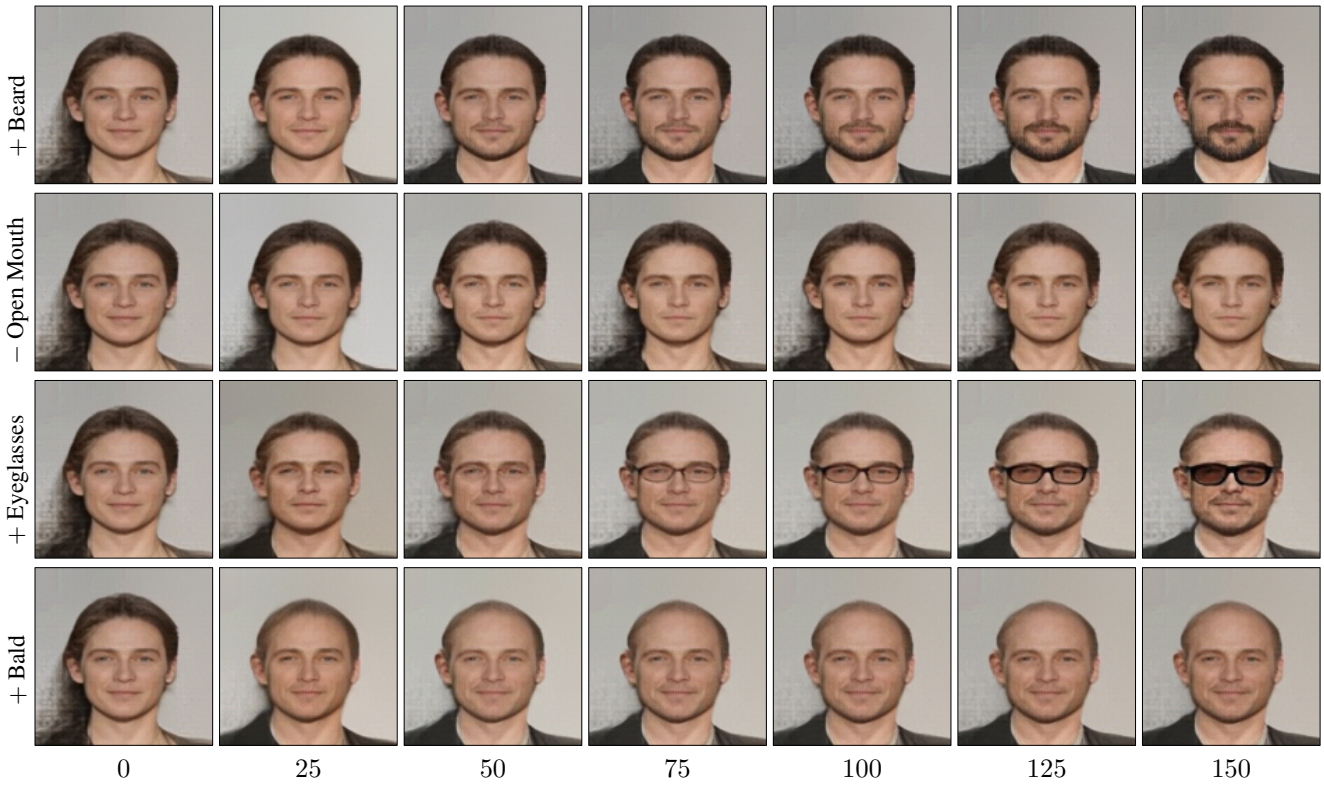

Figure C.3. **Change different attributes of a single identity.** We visualize the change of a single identity by visualizing the changes in *different* experiments on the *same* latent vector (the bottom row includes the number of iterations).