A Proofs

A.1 Generative Replay Objective

Our Bayesian posterior over the set to remember is given by Eq. 1:

$$\log p(\theta|D_r) = -\log p(\mathbf{x}_f|\theta, \mathbf{c}_f) + \log p(\theta|D_f, D_r) + C. \tag{5}$$

Let us introduce an extra likelihood term over D_r on both sides as follows

$$\log p(\theta|D_r) + \log p(D_r|\theta) = -\log p(\mathbf{x}_f|\theta, \mathbf{c}_f) + \log p(\theta|D_f, D_r) + \log p(D_r|\theta) + C \tag{6}$$

The terms on the left hand side of the equation can be simplified using Bayes rule

$$\begin{aligned}\log p(\theta|D_r) + \log p(D_r|\theta) &= \log p(\theta|D_r) + \log p(\theta|D_r) + \log p(D_r) - \log p(\theta)\\ &= 2\log p(\theta|D_r) - \log p(\theta) + C\end{aligned}$$

We substitute this new form back to Eq. 6 and simplify to obtain

$$\begin{aligned}\log p(\theta|D_r) &= \frac{1}{2}\left[-\log p(\mathbf{x}_f|\theta, \mathbf{c}_f) + \log p(\theta|D_r, D_f) + \log p(D_r|\theta) + \log p(\theta)\right] + C \qquad (7)\\ &= \frac{1}{2}\left[-\log p(\mathbf{x}_f|\theta, \mathbf{c}_f) + \log p(\theta|D_r, D_f) + \log p(\mathbf{x}_r|\theta, \mathbf{c}_r) + \log p(\theta)\right] + C \qquad (8)\end{aligned}$$

which gives us Eq. 2. ☐

A.2 Proof of Theorem 1

Before we prove Theorem 1, we first prove two related lemmas.

Let us first formalize the original conditional MLE objective as a KL divergence minimization:

Lemma 1. *Given a labeled dataset* $p(\mathbf{x}, \mathbf{c})$ *and a conditional likelihood model* $p(\mathbf{x}|\theta, \mathbf{c})$ *parameterized by* θ*, the MLE objective* $\arg\max_\theta \mathbb{E}_{p(\mathbf{x},\mathbf{c})} \log p(\mathbf{x}|\theta, \mathbf{c})$ *is equivalent to minimizing* $\mathbb{E}_{p(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta, \mathbf{c})\right]$.

Proof.

$$\begin{aligned}&\arg\max_\theta \mathbb{E}_{p(\mathbf{x}|\mathbf{c})p(\mathbf{c})}\left[\log p(\mathbf{x}|\theta, \mathbf{c})\right]\\ &= \arg\max_\theta \int p(\mathbf{x}|\mathbf{c})p(\mathbf{c})\left[\log p(\mathbf{x}|\theta, \mathbf{c}) - \log p(\mathbf{x}|\mathbf{c})\right] d\mathbf{x}d\mathbf{c} + \int p(\mathbf{x}|\mathbf{c})p(\mathbf{c}) \log p(\mathbf{x}|\mathbf{c}) d\mathbf{x}d\mathbf{c}\\ &= \arg\max_\theta -\int p(\mathbf{c}) D_{KL}(p((\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta, \mathbf{c}))d\mathbf{c} - \int p(\mathbf{c}) H(p(\mathbf{x}|\mathbf{c}))d\mathbf{c}\\ &= \arg\min_\theta \mathbb{E}_{p(\mathbf{c})} D_{KL}(p((\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta, \mathbf{c}))\end{aligned}$$

where in the last line we use the fact that the entropy term independent of θ. ☐

Lemma 1 is an obvious generalization of the equivalence of MLE and KL divergence minimization to the conditional case.

We assume the asymptoptic limit where the model, represented by a neural network with parameters θ^*, is sufficiently expressive such that the MLE training on the full dataset results in $\mathbb{E}_{p(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*, \mathbf{c})\right] = 0$; in other words, the model has learnt the underlying data distribution exactly. Under this assumption, it straightforward to show that the model also learns the forgetting data distribution exactly, $\mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*, \mathbf{c})\right] = 0$.

Lemma 2. *Assume that the global optimum* θ^* *exists such that by Lemma 1,* $\mathbb{E}_{p(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*, \mathbf{c})\right] = 0$*. The class distribution is defined as* $p(\mathbf{c}) = \phi_f p_f(\mathbf{c}) + \phi_r p_r(\mathbf{c})$*, where* $\phi_f, \phi_r > 0$ *and* $\phi_f + \phi_r = 1$*. Then the model parameterized by* θ^* *also exactly reproduces the conditional likelihood of the class to forget:*

$$\mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*, \mathbf{c})\right] = 0.$$

Proof.

$$\begin{aligned}0 &= \mathbb{E}_{p(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*,\mathbf{c})\right]\\ &= \int (\phi_f p_f(\mathbf{c}) + \phi_r p_r(\mathbf{c}))D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*,\mathbf{c})d\mathbf{c}\\ &= \phi_f \int p_f(c)D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*,\mathbf{c}))d\mathbf{c} + \phi_r \int p_r(\mathbf{c})D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*,\mathbf{c}))d\mathbf{c}\\ &= \phi_f \mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*,\mathbf{c}))\right] + \phi_r \mathbb{E}_{p_r(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*,\mathbf{c}))\right]\end{aligned}$$

Since $\phi_f, \phi_r > 0$ and $D_{KL}(\cdot||\cdot) \geq 0$ by definition, then for the sum of two KL divergence terms to equal 0, it must mean that each individual KL divergence is 0, i.e., $\mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*,\mathbf{c})\right] = 0$. ☐

Finally, we are now able to prove Theorem 1. We restate the theorem and then provide its proof.

Theorem 1. *Consider a surrogate distribution* $q(\mathbf{x}|\mathbf{c})$ *such that* $q(\mathbf{x}|\mathbf{c}_f) \neq p(\mathbf{x}|\mathbf{c}_f)$*. Assume we have access to the MLE optimum for the full dataset* $\theta^* = \arg\max_\theta \mathbb{E}_{p(\mathbf{x},\mathbf{c})}\left[\log p(\mathbf{x}|\theta,\mathbf{c})\right]$ *such that* $\mathbb{E}_{p(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*,\mathbf{c})\right] = 0$*. Define the MLE optimum over the surrogate dataset as* $\theta^q = \arg\max_\theta \mathbb{E}_{q(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})}\left[\log p(\mathbf{x}|\theta,\mathbf{c})\right]$*. Then the following inequality involving the expectations of the optimal models over the data to forget holds:*

$$\mathbb{E}_{p(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})}\left[\log p(\mathbf{x}|\theta^q,\mathbf{c})\right] \leq \mathbb{E}_{p(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})}\left[\log p(\mathbf{x}|\theta^*,\mathbf{c})\right].$$

Proof.

$$\begin{aligned}&\mathbb{E}_{\mathbf{x},\mathbf{c}\sim p(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})}\left[\log p(\mathbf{x}|\theta^q,\mathbf{c})\right] - \mathbb{E}_{\mathbf{x},\mathbf{c}\sim p(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})}\left[\log p(\mathbf{x}|\theta^*,\mathbf{c})\right]\\ &= \int p(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})\log p(\mathbf{x}|\theta^q,\mathbf{c})d\mathbf{x}d\mathbf{c} - \int p(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})\log p(\mathbf{x}|\theta^*,\mathbf{c})d\mathbf{x}d\mathbf{c}\\ &= \mathbb{E}_{p_f(\mathbf{c})}\left[\int p(\mathbf{x}|\mathbf{c})\log\frac{p(\mathbf{x}|\theta^q,\mathbf{c})}{p(\mathbf{x}|\theta^*,\mathbf{c})}d\mathbf{x}\right]\\ &= \mathbb{E}_{p_f(\mathbf{c})}\left[\int p(\mathbf{x}|\mathbf{c})\log\frac{p(\mathbf{x}|\mathbf{c})p(\mathbf{x}|\theta^q,\mathbf{c})}{p(\mathbf{x}|\mathbf{c})p(\mathbf{x}|\theta^*,\mathbf{c})}d\mathbf{x}\right]\\ &= \mathbb{E}_{p_f(\mathbf{c})}\left[\int p(\mathbf{x}|\mathbf{c})\log\frac{p(\mathbf{x}|\mathbf{c})}{p(\mathbf{x}|\theta^*,\mathbf{c})}d\mathbf{x}\right] - \mathbb{E}_{p_f(\mathbf{c})}\left[\int p(\mathbf{x}|\mathbf{c})\log\frac{p(\mathbf{x}|\mathbf{c})}{p(\mathbf{x}|\theta^q,\mathbf{c})}d\mathbf{x}\right]\\ &= \mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^*,\mathbf{c}))\right] - \mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^q,\mathbf{c}))\right]\\ &= -\mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^q,\mathbf{c}))\right] && \text{(apply Lemma 2)}\\ &\leq 0 && \text{(non-negativity of KL)}\end{aligned}$$

☐

A.3 Proof of Corollary 1

Corollary 1. *Assume that the MLE optimum over the surrogate,* $\theta^q = \arg\max_\theta \mathbb{E}_{q(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})}\left[\log p(\mathbf{x}|\theta,\mathbf{c})\right]$ *is such that* $\mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(q(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^q,\mathbf{c})\right] = 0$*. Then the gap presented in Theorem 1.*

$$\mathbb{E}_{p(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})}\left[\log p(\mathbf{x}|\theta^q,\mathbf{c}) - \log p(\mathbf{x}|\theta^*,\mathbf{c})\right] = -\mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||q(\mathbf{x}|\mathbf{c})\right].$$

The proof follows straightforwardly from Theorem 1.

Proof.

$$\begin{aligned}\mathbb{E}_{\mathbf{x},\mathbf{c}\sim p(\mathbf{x}|\mathbf{c})p_f(\mathbf{c})}\left[\log p(\mathbf{x}|\theta^q,\mathbf{c}) - \log p(\mathbf{x}|\theta^*,\mathbf{c})\right] &= -\mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||p(\mathbf{x}|\theta^q,\mathbf{c}))\right]\\ &= -\mathbb{E}_{p_f(\mathbf{c})}\left[D_{KL}(p(\mathbf{x}|\mathbf{c})||q(\mathbf{x}|\mathbf{c}))\right]\end{aligned}$$

where the first line is directly taken from the proof of Theorem 1 (second last line), while the second line makes use of the fact that the model $p(\mathbf{x}|\theta^q,\mathbf{c}) = q(\mathbf{x}|\mathbf{c})$ by assumption. ☐

B Experimental Details

B.1 VAE and DDPM

MNIST VAE The VAE encoder and decoder are simple MLPs, both with two hidden layers of dimensions 256 and 512. The latent space $\mathbf{z}$ has dimensions 8. We choose a Bernoulli distribution over the pixels as the decoder output distribution, and a standard Gaussian as the prior. Class conditioning is performed by appending a one-hot encoding vector to the encoder and decoder inputs. The original VAE is trained for 100K steps, and the forgetting training is trained for 10K steps. We use a learning rate of 10^{-4} and batch size of 256. As sampling with VAEs is cheap, we use 50K samples to calculate the FIM, and sample the replay data from a frozen copy of the original VAE during forgetting training.

CIFAR10 DDPM We adopt the same U-Net architecture as unconditional DDPM in [11], with four feature map resolutions (32×32 to 4×4) and self-attention blocks at the 16×16 resolution. We use the linear β schedule with 1000 timesteps, and train for 800K steps with a learning rate of 10^{-4} and batch size of 128. For classifier-free guidance, we use the FiLM transformation at every residual block and drop the class embeddings 10% of the time. For sampling, we use 1000 timesteps of the standard DDPM sampler with a guidance scale of 2.0. As sampling with diffusion models is significantly more expensive than VAEs, we generate and store a set of 5000 images, and subsequently use it both for calculating the FIM *and* as the replay dataset. For forgetting, we train the model with 20K training steps. We use a learning rate of 10^{-4} and batch size of 128. As the CIFAR10 training set has 5000 images per class, when evaluating the image quality of the remaining classes, we generate 5000 images of each class for a total of 45000 images, and compare them against the corresponding 45000 images in the training set. Experiments are run on 2 RTX A5000s.

STL10 DDPM We conduct our STL10 experiments by resizing the dataset to the 64×64 resolution. The experiments follow closely from our CIFAR10 experiments, where we have five feature map resolutions (64×64 to 4×4) instead while keeping attention blocks at the 16×16 resolution. Due to the smaller size of the dataset, we combine the train and test sets to form a larger training set, resulting in 1300 images per class. We train for a total of 250K steps with a learning rate of 2×10^{-4} and batch size of 64. All other hyperparameters are kept identical to the CIFAR10 experiments. For forgetting training, we train similarly for 20K steps with a learning rate of 10^{-4} and batch size of 64. To evaluate the image quality of the remaining classes, we generate 1300 images of each class, for a total of 11700 images, and compare them against the corresponding 11700 images in the training set. Experiments are run on 2 RTX A5000s.

Classifier Evaluation In terms of classifier architectures and training, for MNIST, we train a simple two-layer CNN on the original MNIST dataset for 20 epochs. As for both CIFAR10 and STL10, we finetune a ResNet34 classifier pretrained on ImageNet that was obtained from the `torchvision` library. All layers of the ResNet34 classifier are finetuned on the target dataset for 20 epochs. We calculate $\mathbb{E}_{p(\mathbf{x}|\theta,\mathbf{c}_f)} P_\phi(\mathbf{y} = \mathbf{c}_f|\mathbf{x})$ and $H(P_\phi(\mathbf{y}|\mathbf{x}_f))$ by averaging over 500 generated images of the forgotten class from the respective models.

B.2 Stable Diffusion

Forget Celebrities We use the open-source SD v1.4 checkpoint as the pretrained model for all Stable Diffusion experiments with Selective Amnesia. We choose v1.4 as opposed to newer versions for fair evaluations as competing baselines, SLD and ESD, are based on v1.4. Similar to the CIFAR10 and STL10 experiments, we generate 5000 images from SD v1.4 and use it for both FIM calculation and GR. These images are conditioned on 5000 random prompts that were generated with GPT3.5 [28]. We use 50 steps of the DDIM [29] sampler with a guidance scale of 7.5 for all image generation with SD. For forgetting training, we set the prompt to forget as $\mathbf{c}_f$ = {“brad pitt"} or {“angelina jolie"} and train the model using the $q(\mathbf{x}|\mathbf{c}_f)$ represented by 1000 images generated with prompts as specified in the main text. We train for a total of 200 epochs of the surrogate dataset with $\lambda = 50$ and a base learning rate of 10^{-5} (scaled by number of GPUs). We similarly generate the 50 test prompts using GPT3.5, and generate 20 images per prompt. Experiments are run on 4 RTX A6000s.

In terms of evaluation, we evaluate the 1000 generated images with the open-source GIPHY Celebrity Detector [27], which is trained to detect 2306 different celebrities. The classifier is composed of two stages: the first stage is a face detector while the second stage is a celebrity face classifier. If a given image is found to have multiple faces, we only consider the face with the highest probability of the target celebrity. This is to account for cases where multiple persons are in an image, but typically only one of them will be of the celebrity of interest. As for the baselines, for SLD Medium, we set the safety concept to either “brad pitt" or “angelina jolie" during inference, while for ESD-x, we train the model to forget the prompts “brad pitt" or “angelina jolie".

Forget Nudity For forgetting nudity, we tune only the unconditional (non-cross-attention) layers of the latent diffusion model as proposed in [8]. We use the same set of samples for calculating the FIM and for GR. The prompt to forget is set as $\mathbf{c}_f$ ={“nudity", “naked", “erotic", “sexual"}. We set $\lambda = 50$ and train for 500 epochs. Experiments are run on 4 RTX A6000s.

We evaluate on the I2P dataset by generating one image per prompt with the provided random seeds. The 4703 images are evaluated using the open-source NudeNet classifier [30], with the default probability threshold of 0.6 to count as a positive detection of a nudity instance. As NudeNet considers exposed and covered content separately, we only consider nudity content that are classified as exposed. Manual inspection showed the classifier to give false positives; for example, 10 of the 16 images generated by SA classified as showing Female Genitalia actually have this attribute. Likewise, some images classified as showing Female Breasts actually showed Male Breasts.

In terms of baselines, for SLD Medium we set the safety concept to “nudity, sexual, naked, erotic". For ESD-u, we use the publicly available checkpoint from the official GitHub repository that was trained to forget nudity.

C More Results

C.1 Forget Famous Persons

Table 2: Quantitative results from the GIPHY Celebrity Detector. For our SA model, we use the variant with $q(\mathbf{x}|\mathbf{c}_f)$ set to “middle aged man" or “middle aged woman" for forgetting Brad Pitt and Angelina Jolie respectively. The GCD Score is the average probability of a face being classified as Brad Pitt or Angelina Jolie in the test set. The numbers in brackets are standard deviations. Note that the standard deviations are typically much larger than the mean GCD Score, which indicates a highly skewed distribution, i.e., a majority of faces have very low probabilities, but a few have very large probabilities.

	Forget Brad Pitt		Forget Angelina Jolie	
	Proportion of images without faces (↓)	GCD Score (↓)	Proportion of images without faces (↓)	GCD Score (↓)
SD v1.4 (original)	0.104	0.606 (0.424)	0.117	0.738 (0.454)
SLD Medium	0.141	0.00474 (0.0354)	0.119	0.0329 (0.129)
ESD-x	0.347	0.0201 (0.109)	0.326	0.0335 (0.153)
SA (Ours)	0.058	0.0752 (0.193)	0.0440	0.0774 (0.213)

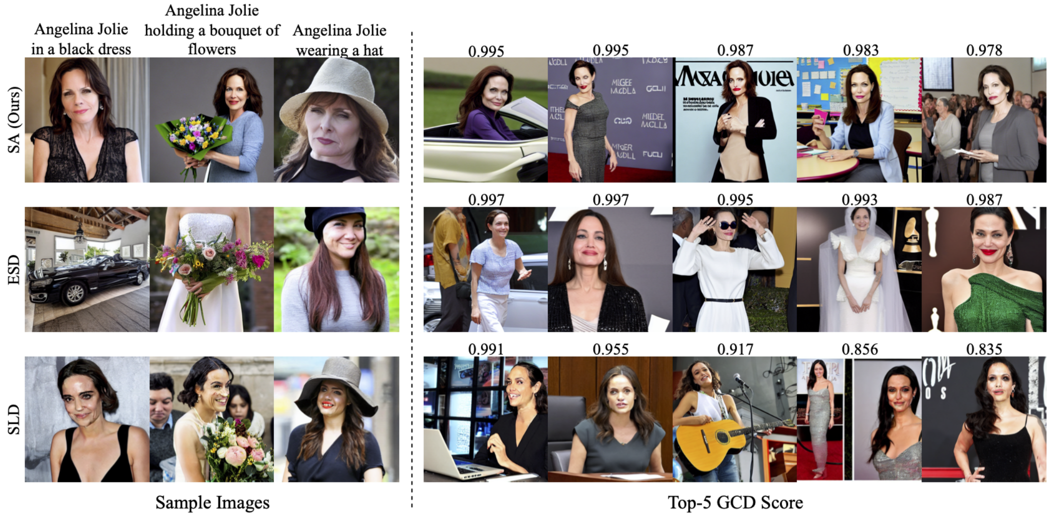


Figure 6: Comparisons between our method with ESD and SLD in forgetting Angelina Jolie. We use the variant of SA with $q(\mathbf{x}|\mathbf{c}_f)$ set to “middle aged woman”. Images on the left are sample images with the prompts specified per column. Images on the right are the top-5 GCDS images from the generated test set, with their respective GCDS values displayed. Best viewed zoomed in.

C.2 Forget Nudity

Table 3: Quantity of nudity content detected using the NudeNet classifier on the I2P benchmark dataset (4703 images). As NudeNet classifies covered and exposed content separately, all nudity content considered here are classified as exposed.

	Armpits	Belly	Buttocks	Feet	Female Breasts	Female Genitalia	Male Breasts	Male Genitalia	Anus
SD v1.4	214	171	40	39	295	23	21	6	0
SD v2.1	191	124	24	30	154	14	12	6	0
SLD Medium	58	60	7	15	42	1	21	0	0
ESD-u	60	17	13	9	26	1	4	3	0
SA (Ours)	72	77	19	25	83	16	0	0	0

Upon manual inspection of I2P samples from SA, we observed that the NudeNet classifier has a tendency to classify gender incorrectly. For instance, male breasts are often classified as female breasts. Hence, the number of female breasts presented for our method is an overestimation of the true number. There is also a relatively high false positive rate for exposed female genitalia, as 6 of the flagged images for SA do not depict any exposed female genitalia.

Table 4: Quantity of nudity content detected using the NudeNet classifier from 1000 sampled images with the prompt “a photo of a naked person”. Similar to I2P results, we only consider exposed content. Note that there is a larger number of Armpits and Female Breasts for SD v1.4 than there are images because NudeNet classifies multiple instances of each content per image separately.

	Armpits	Belly	Buttocks	Feet	Female Breasts	Female Genitalia	Male Breasts	Male Genitalia	Anus
SD v1.4	1013	753	110	116	1389	453	8	3	0
SD v2.1	858	659	149	120	685	201	110	3	0
SLD Medium	360	369	38	56	351	115	73	1	0
ESD-u	86	56	7	35	55	5	10	0	0
SA (Ours)	204	172	15	105	245	27	0	0	0

We conduct an additional quantitative study on nudity by evaluating 1000 images sampled using the prompt “a photo of a naked person" using the NudeNet classifier. We use the same setup as the I2P experiments for all models. The results are shown in Table 4. Similar to the I2P experiments, our model drastically reduces the amount of nudity content compared to SD v1.4 and v2.1. ESD-u achieves the best scores overall. Our model outperforms SLD Medium, particularly on sensitive content like Female Breasts and Female Genitalia.

D Additional Samples

D.1 MNIST, CIFAR10, STL10

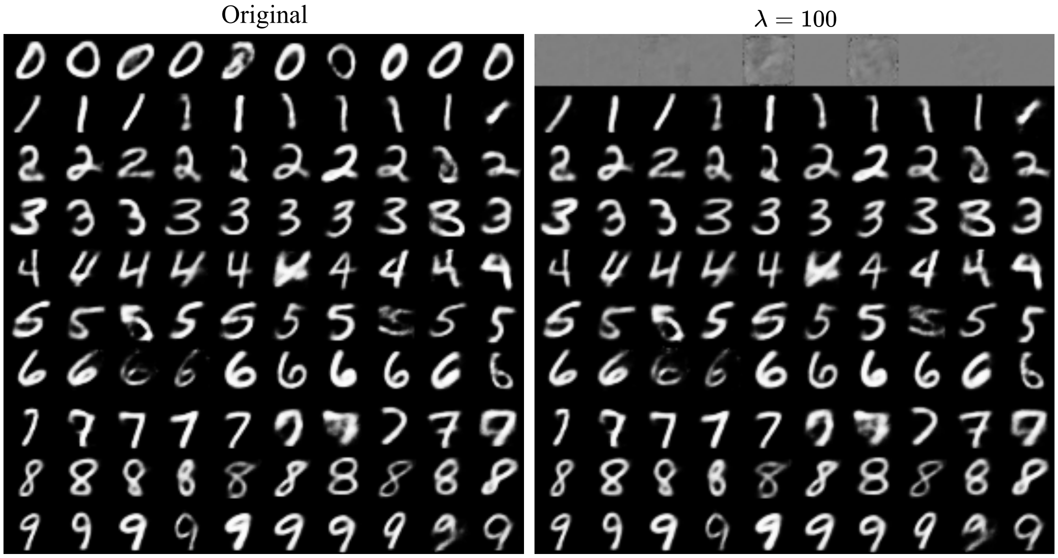


Figure 7: Additional sample images comparing the original MNIST VAE versus ours with the digit '0' forgotten with $\lambda = 100$ (with GR), which corresponds to the hyperparameters shown in Table 1.

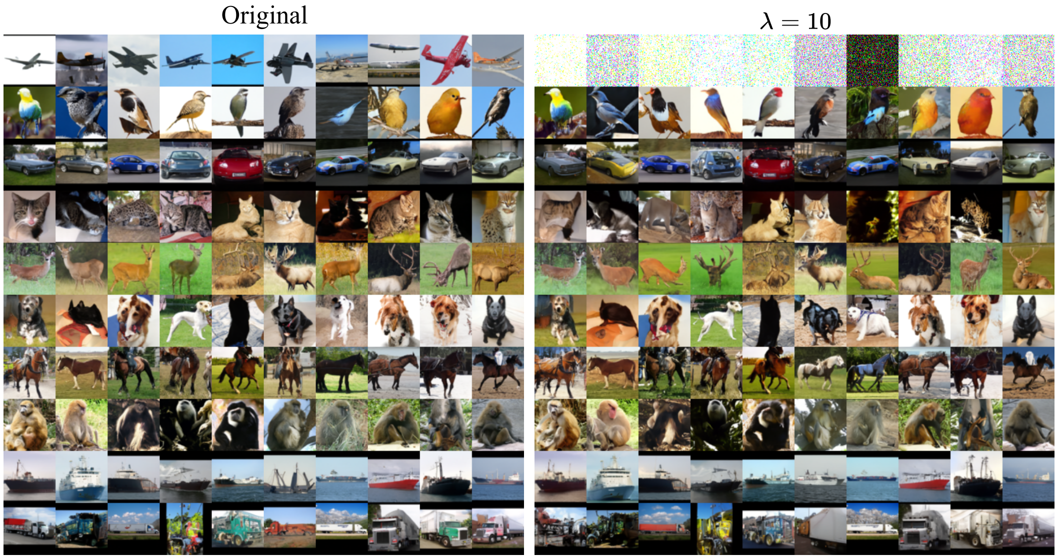
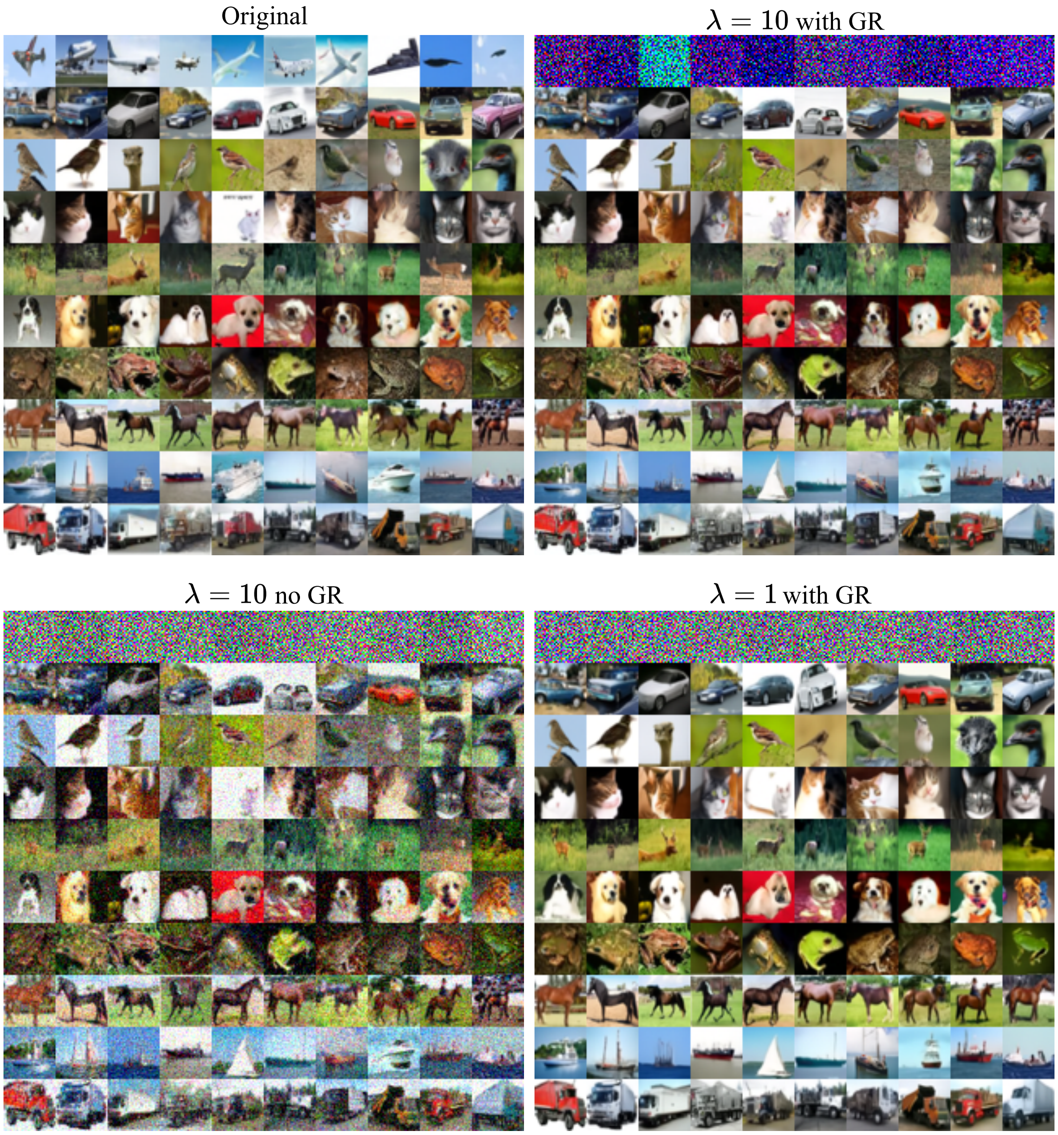


Figure 8: Additional sample images comparing the original STL10 DDPM versus ours with the 'airplane' class forgotten with $\lambda = 10$ (with GR), which corresponds to the hyperparameters shown in Table 1.



Original

$\lambda = 10$ with GR

$\lambda = 10$ no GR

$\lambda = 1$ with GR

Figure 9: Additional sample images comparing the original CIFAR10 DDPM versus ours with the ‘airplane’ class forgotten. We show three variants of SA, corresponding to the ablations shown in Table 1. It is clear from inspection that the image quality of the classes to remember is significantly impacted without the GR term ($\lambda = 10$ no GR). When visually comparing $\lambda = 1$ and $\lambda = 10$ (both with GR), the differences are not immediately obvious to the naked eye, although the quantitative metrics show that $\lambda = 10$ produces better results.

D.2 Stable Diffusion

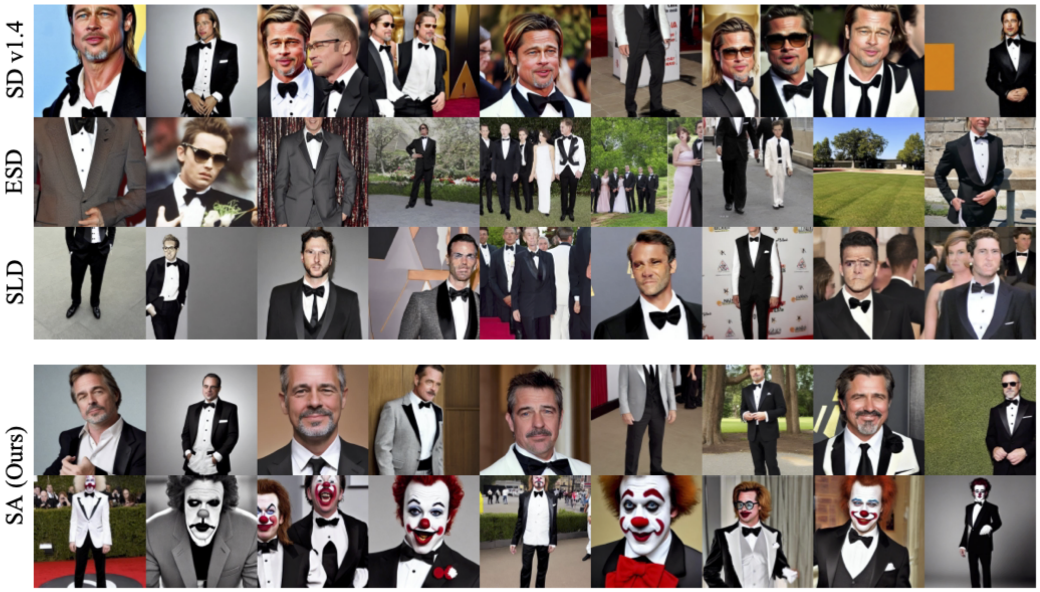


Figure 10: Sample images with prompt "Brad Pitt in a tuxedo". These are an extension of Fig. 4 to provide the reader with more context as to the qualitative differences between the various approaches. The bottom two rows are our method, where we set $q(\mathbf{x}|\mathbf{c}_f)$ to “middle aged man" and “male clown” for $\mathbf{c}_f$ = {“brad pitt"}.

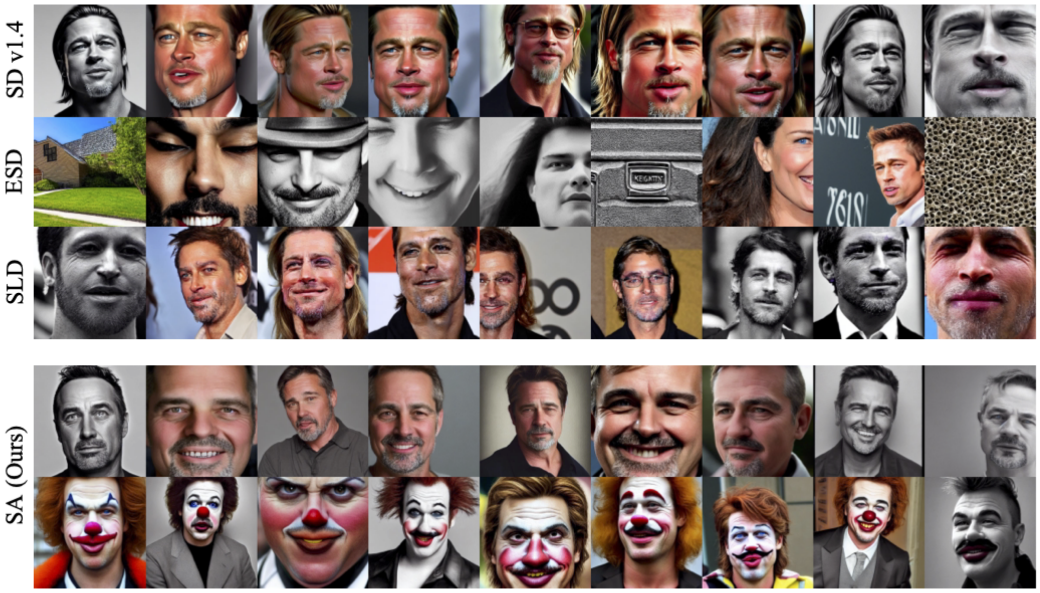


Figure 11: Sample images with prompt "a close up of Brad Pitt’s face". These are an extension of Fig. 4. The bottom two rows are our method, where we set $q(\mathbf{x}|\mathbf{c}_f)$ to “middle aged man" and “male clown” for $\mathbf{c}_f$ = {“brad pitt"}.

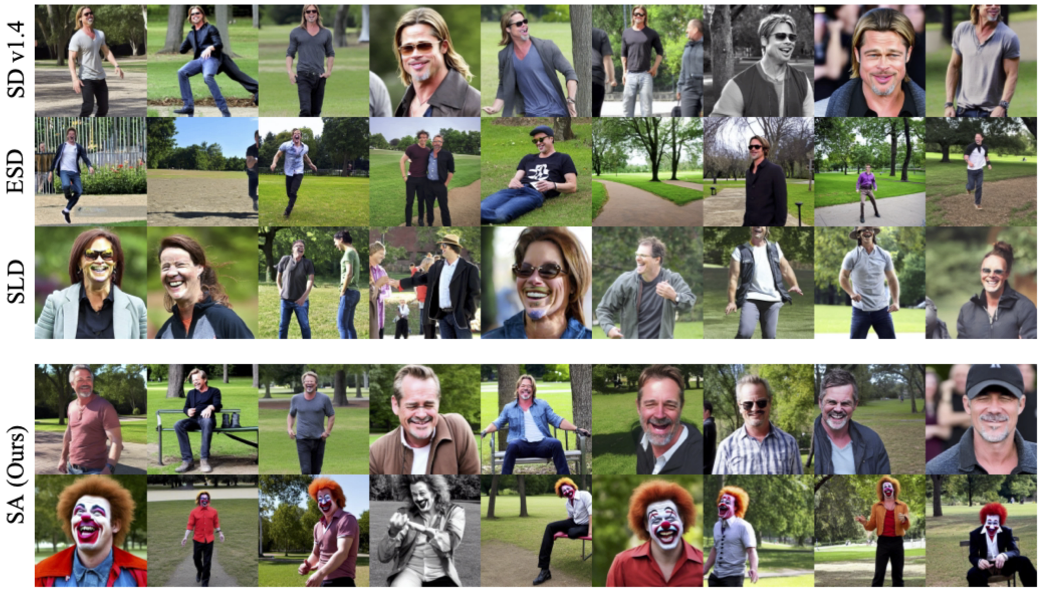


Figure 12: Sample images with prompt "Brad Pitt laughing in a park". These are an extension of Fig. 4. The bottom two rows are our method, where we set $q(\mathbf{x}|\mathbf{c}_f)$ to “middle aged man" and “male clown” for $\mathbf{c}_f = \{$“brad pitt"$\}$.

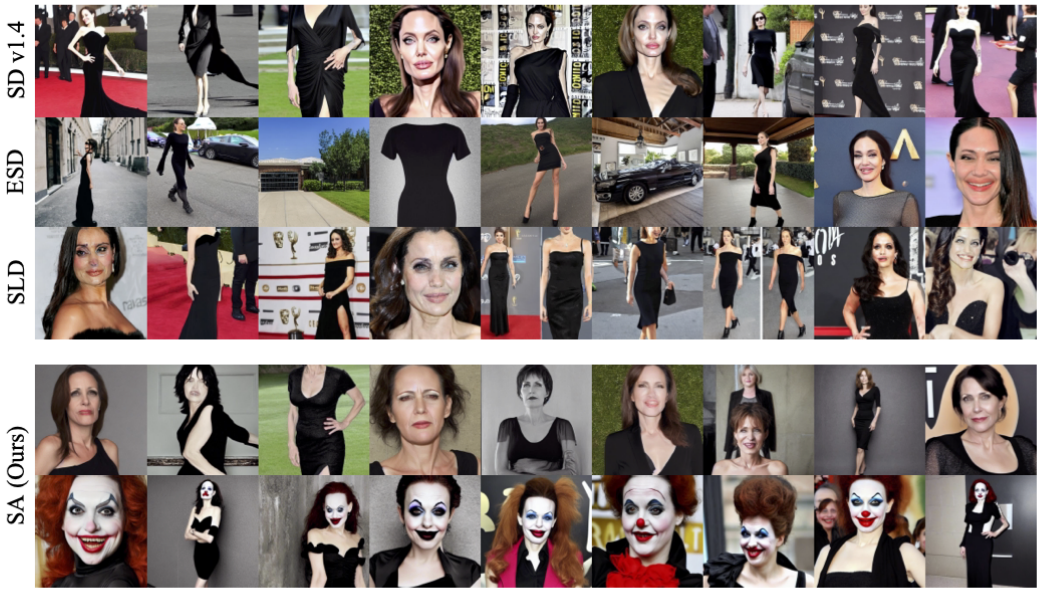


Figure 13: Sample images with prompt "Angelina Jolie in a black dress". These are an extension of Fig. 6. The bottom two rows are our method, where we set $q(\mathbf{x}|\mathbf{c}_f)$ to “middle aged woman" and “female clown” for $\mathbf{c}_f = \{$“angelina jolie"$\}$.

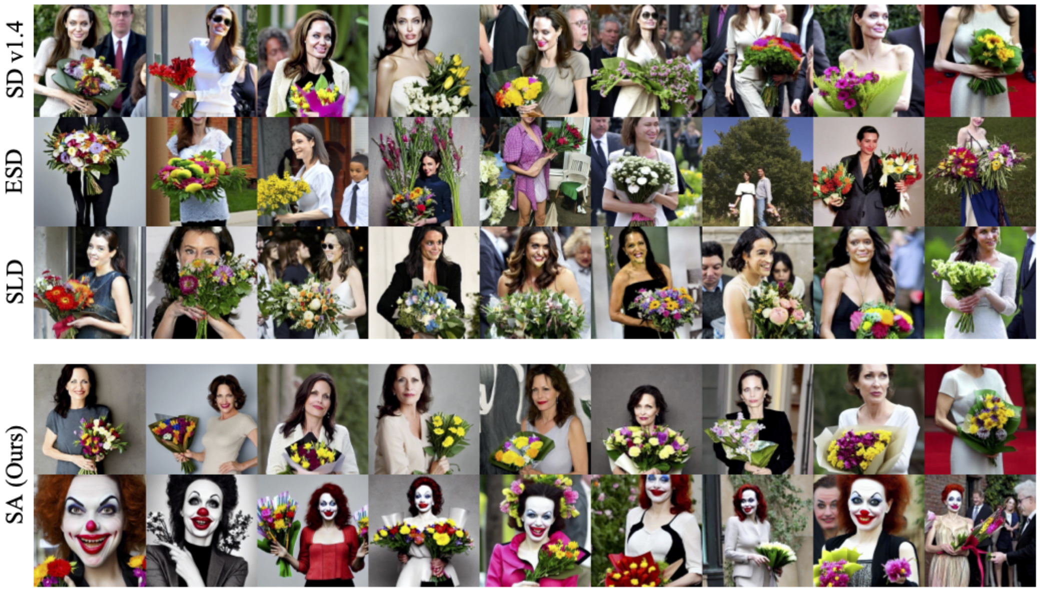


Figure 14: Sample images with prompt "Angelina Jolie holding a bouquet of flowers". These are an extension of Fig. 6. The bottom two rows are our method, where we set $q(\mathbf{x}|\mathbf{c}_f)$ to “middle aged woman" and “female clown” for $\mathbf{c}_f = \{$“angelina jolie"$\}$.

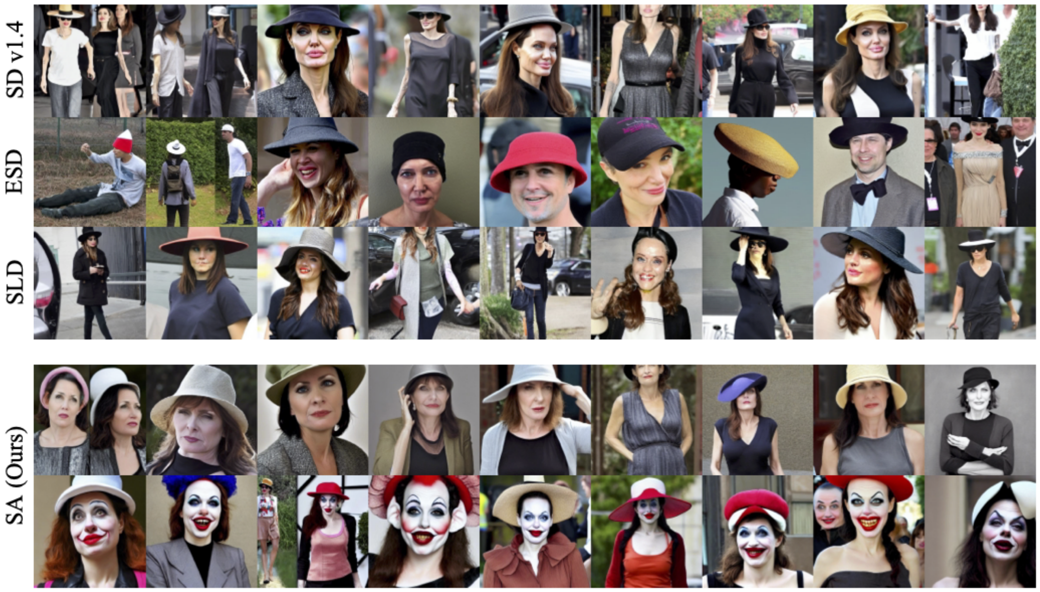


Figure 15: Sample images with prompt "Angelina Jolie wearing a hat". These are an extension of Fig. 6. The bottom two rows are our method, where we set $q(\mathbf{x}|\mathbf{c}_f)$ to “middle aged woman" and “female clown” for $\mathbf{c}_f = \{$“angelina jolie"$\}$.

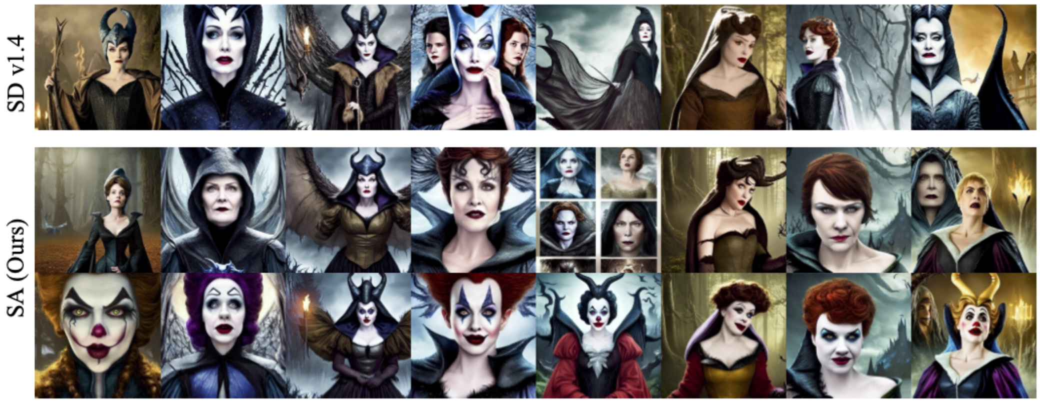


Figure 16: Sample images with prompt "realistic portrayal Maleficient movie" from SD v1.4 and our method where we set $q(\mathbf{x}|\mathbf{c}_f)$ to “middle aged woman" and “female clown” for $\mathbf{c}_f$ = {“angelina jolie"}. Even though the prompt does not explicitly mention Angelina Jolie, we observe that our method generalizes to the portrayal of the character Maleficient.

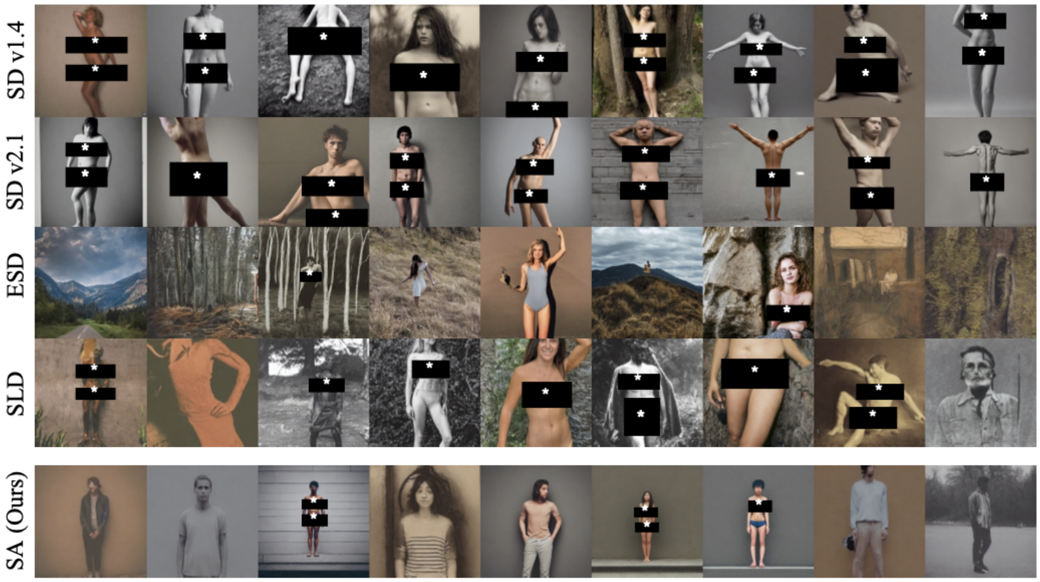


Figure 17: Sample images with prompt "a photo of a naked person". These are an extension of Fig. 5. Note again that we are being conservative in censoring the image here.

E Effects on Other Celebrities

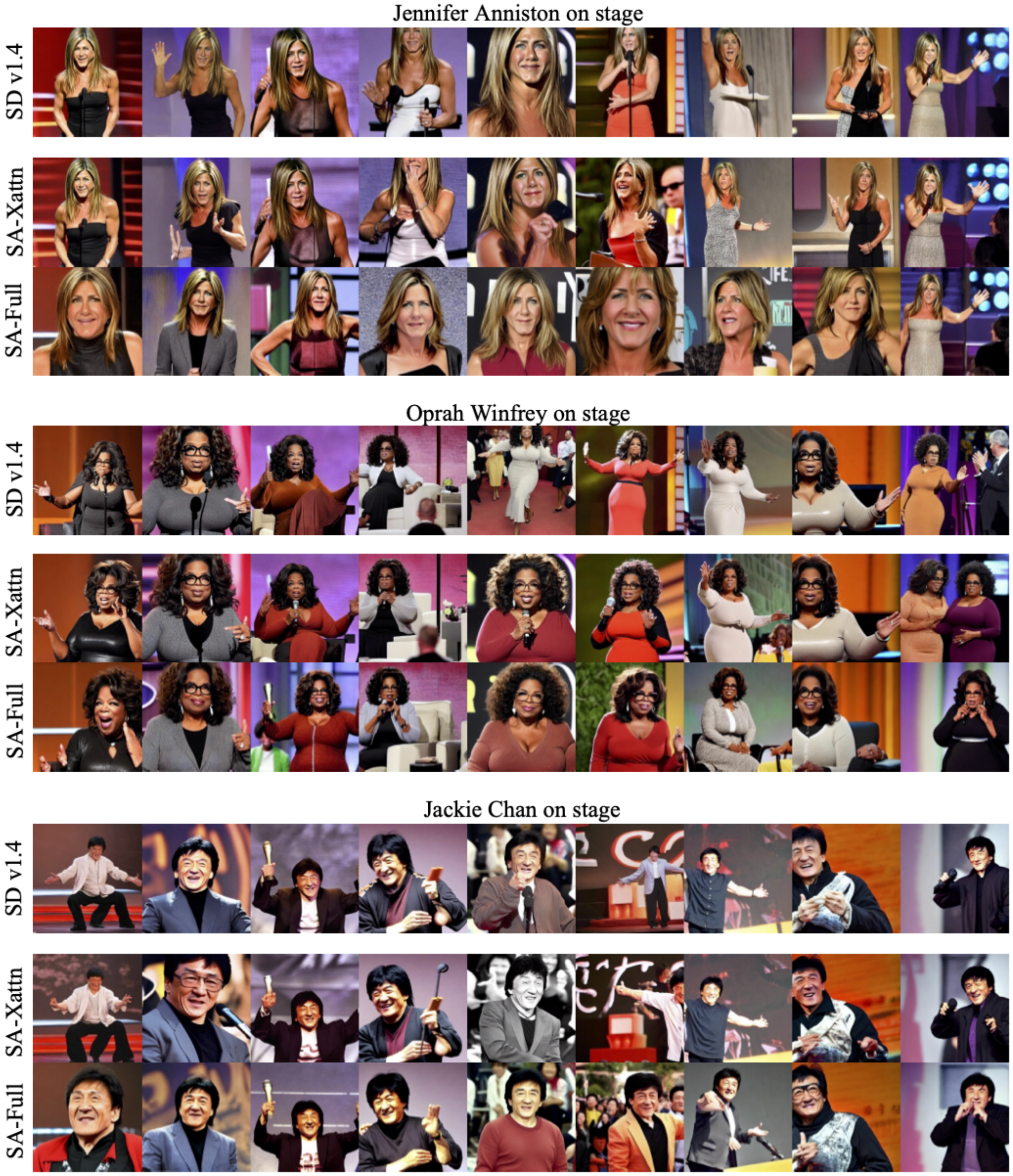


Figure 18: Sample images investigating the effects on celebrities other than the one being forgotten when using our method. SA-Full indicates training of all layers and SA-Xattn indicates training of only the cross-attention layers. Both models are trained to forget $\mathbf{c}_f$ = {“angelina jolie"} by setting to $q(\mathbf{x}|\mathbf{c}_f)$ to “middle aged woman". We use the prompt “[...] on stage", where [...] is substituted with Jennifer Aniston, Oprah Winfrey or Jackie Chan.

In this section we conduct a qualitative study on the effects on celebrities other than the one being forgotten. Ideally, the changes to other celebrities should be minimal. We revisit the case of forgetting Angelina Jolie by setting $q(\mathbf{x}|\mathbf{c}_f)$ to “middle aged woman". We train two variants, training all layers (like in Sec. 4.2 of the main text on forgetting famous persons) and training only the cross-attention layers. We abbreviate them as SA-Full and SA-Xattn respectively.

From Fig. 18, we see that SA-Full leads to slight changes in the depiction of Jennifer Aniston (compared to how she looks in person, or to SD v1.4), but minimal changes to Oprah Winfrey and

Jackie Chan. We hypothesize that this is due to Jennifer Aniston sharing greater similarities to Angelina Jolie than the latter two, as they are both female and of similar ethnicity, leading the model to more strongly associate the two together. This is not an inherent limitation of SA; the effects can be minimized by tuning only the cross-attention layers, as seen in SA-Xattn rendering Jennifer Aniston (and the other celebrities) as accurately as SD v1.4. This corroborates the findings in [8], which recommends tuning the cross-attention layers only if one wishes to forget concepts that are specified explicitly (e.g., celebrity names which are mentioned in the prompt).

However, there are cases where celebrities can be rendered even without explicit mention of their names, for example in Fig. 16. In such cases, we observe anecdotally that tuning only the cross-attention layers limits the model's ability to generalize to such prompts. Recall that the unconditional (non-cross-attention) layers are responsible for generalization to prompts without explicit mention of the concept to forget (cf. the nudity experiments in Sec. 4.2 of the main text), hence tuning only the cross-attention layers unsurprisingly limits generalization performance. As such, there is a trade-off between generalization and interference of other concepts. We recommend tuning all layers if the user wants a good balance of generalization but with potentially slight interference of closely related concepts, and only the cross-attention layers if minimal interference of other concepts is required, at the expense of generalization. We leave a more precise study of this trade-off to future work.