Erasing Undesirable Influence in Diffusion Models

Supplementary Material

7. Proofs

Theorem 3.1 The optimal solution of the optimization problem in Eq. (6) is $\boldsymbol{\delta}^* = \nabla_{\boldsymbol{\theta}}\mathcal{L}_r(\boldsymbol{\theta}_t;\mathcal{D}_r) + \lambda_t \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t)$ where $\lambda_t = \max\{0, \frac{a_t - \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t)^\top \nabla_{\boldsymbol{\theta}}\mathcal{L}_r(\boldsymbol{\theta}_t;\mathcal{D}_r)}{\|\nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t)\|_2^2}\}$.

Proof. The Lagrange function with $\lambda \geq 0$ for Eq. (6) is

$$h(\boldsymbol{\delta}, \lambda) = \frac{1}{2}\left\|\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_t;\mathcal{D}_r) - \boldsymbol{\delta}\right\|_2^2 + \lambda(a_t - \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t)^\top \boldsymbol{\delta}). \tag{7}$$

Then, using the Karush-Kuhn-Tucker (KKT) theorem, at the optimal solution we have

$$\begin{aligned} \boldsymbol{\delta} - \nabla_{\boldsymbol{\theta}}\mathcal{L}_r(\boldsymbol{\theta}_t;\mathcal{D}_r) - \lambda \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t) &= \mathbf{0}, \\ \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t)^\top \boldsymbol{\delta} &\geq a_t, \\ \lambda(a_t - \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t)^T \boldsymbol{\delta}) &= 0, \\ \lambda &\geq 0. \end{aligned} \tag{8}$$

From the above constraints, we can obtain:

$$\begin{aligned} \boldsymbol{\delta} &= \nabla_{\boldsymbol{\theta}}\mathcal{L}_r(\boldsymbol{\theta}_t;\mathcal{D}_r) + \lambda \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t), \\ \lambda &= \max\{0, \frac{a_t - \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t)^\top \nabla_{\boldsymbol{\theta}}\mathcal{L}_r(\boldsymbol{\theta}_t;\mathcal{D}_r)}{\|\nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}_t)\|_2^2}\}. \end{aligned} \tag{9}$$

□

Theorem 3.2 [Pareto optimality] The stationary point obtained by our algorithm is Pareto optimal of the problem $\min_{\boldsymbol{\theta}}[\mathcal{L}_r(\boldsymbol{\theta};\mathcal{D}_r), \mathcal{L}_f(\boldsymbol{\theta};\mathcal{D}_f)]$.

Proof. Let $\boldsymbol{\theta}^*$ be the solution to our problem. Recall that for the current $\boldsymbol{\theta}$, we find $\boldsymbol{\phi}^K$ to minimize $g(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathcal{L}_f(\boldsymbol{\theta};\mathcal{D}_f) - \min \mathcal{L}_f(\boldsymbol{\phi};\mathcal{D}_f)$. Assume that we can update in sufficient number of steps K so that $\boldsymbol{\phi}^K = \boldsymbol{\phi}^*(\boldsymbol{\theta}) = \operatorname{argmin}_{\boldsymbol{\phi}} g(\boldsymbol{\theta}, \boldsymbol{\phi}) = \operatorname{argmin}_{\boldsymbol{\phi}} \mathcal{L}_f(\boldsymbol{\phi};\mathcal{D}_f)$. Here $\boldsymbol{\phi}$ is initialized at $\boldsymbol{\theta}$.

The objective aims to minimize $\mathcal{L}_r(\boldsymbol{\theta};\mathcal{D}_r) + \lambda g(\boldsymbol{\theta};\boldsymbol{\phi}^*(\boldsymbol{\theta}))$, let $\boldsymbol{\theta}^*$ be the optimal solution to this objective. Note that $g(\boldsymbol{\theta}, \boldsymbol{\phi}^*(\boldsymbol{\theta})) = \mathcal{L}_f(\boldsymbol{\theta};\mathcal{D}_f) - \min \mathcal{L}_f(\boldsymbol{\phi}^*(\boldsymbol{\theta});\mathcal{D}_f) \geq 0$ as $\boldsymbol{\phi}$ starts from $\boldsymbol{\theta}$ and is update to decreas $\mathcal{L}_f(\boldsymbol{\phi};\mathcal{D}_f)$. This will decrease to 0 for minimizing the above objective. Therefore, at the optimal solution $\boldsymbol{\theta}^*$, we have $g(\boldsymbol{\theta}^*, \boldsymbol{\phi}^*(\boldsymbol{\theta}^*)) = 0$. This further implies that $\mathcal{L}_f(\boldsymbol{\theta}^*;\mathcal{D}_f) = \min \mathcal{L}_f(\boldsymbol{\phi}^*(\boldsymbol{\theta}^*);\mathcal{D}_f)$, meaning that $\boldsymbol{\theta}^*$ is the current optimal solution of $\mathcal{L}_f(\boldsymbol{\theta}; D_f)$ because we cannot update further the optimal solution. Moreover, we have $\boldsymbol{\theta}^*$ as the local minima of $\mathcal{L}_r(\boldsymbol{\theta};\mathcal{D}_r)$ in sufficiently small vicinity considered, because in the small vicinity around $\boldsymbol{\theta}^*$, $g\left(\boldsymbol{\theta}, \boldsymbol{\phi}^*(\boldsymbol{\theta}^*)\right) = 0$ provides no further improvements for the above sum, any increase in the above objective in the vicinity of $\boldsymbol{\theta}^*$ would primarily be due to an increase in $\mathcal{L}_r(\boldsymbol{\theta};\mathcal{D}_r)$.

□

8. Reproducibility Statement and Details

In this section, we provide detailed instructions on the reproduction of our results, we also share our source code at the repository https://github.com/JingWu321/EraseDiff.

DDPM. Results on conditional DDPM follow the setting in SA [28]. Thanks to the pre-trained DDPM from SA. The batch size is set to be 128, the learning rate is 1×10^{-4}, our model is trained for around 300 training steps. 5K images per class are generated for evaluation. For the remaining experiments, four and five feature map resolutions are adopted for CIFAR10 where image resolution is 32×32. All models apply the linear schedule for the diffusion process. We used A5500 and A100 for all experiments.

SD. We use the open-source SD v1.4 checkpoint as the pre-trained model for all SD experiments. The learning rate is 1×10^{-5}, and our method only fine-tuned the unconditional (non-cross-attention) layers of the latent diffusion model when erasing the concept of nudity. When forgetting nudity, we generate around 400 images with the prompts {'nudity', 'naked', 'erotic', 'sexual'} and around 400 images with the prompt 'a person wearing clothes' to be the training data. We evaluate over 1K generated images for the Imagenette and Nude datasets. 4703 generated images with I2P prompts are evaluated using the open-source NudeNet classifier [2]. The repositories we built upon use the CC-BY 4.0 and MIT Licenses.

9. Additional results

Below, we also provide results on SD for *EraseDiff* when we replace $\boldsymbol{\epsilon}_f$ with $\epsilon_{\boldsymbol{\theta}}(\mathbf{x}_t|c_m)$ like Fan et al. [16], Heng and Soh [28], where c_m is 'a person wearing clothes', denoted as *EraseDiff*$_{\text{wc}}$. The CLIP score and FID score for *EraseDiff*$_{\text{wc}}$ are 30.31 and 19.55, respectively.

To recap, our formulation provides flexibility in choosing $\boldsymbol{\epsilon}_f = \epsilon_{\boldsymbol{\theta}}(\mathbf{x}_t|c_m)$ in Eq.(2), allowing controlled semantic shifts to achieve different levels of content modification. We presented two cases to illustrate this capability: for nudity erasure, setting $c_m =$'a photo of a Pokémon' results in excessive semantic shift, which may lead to blurring. However, $c_m =$'a person wearing clothes' yields a closer match to the original generation while ensuring appropriate modifications. This is indeed a key feature, enabling users to tailor content refinement based on desired constraints.

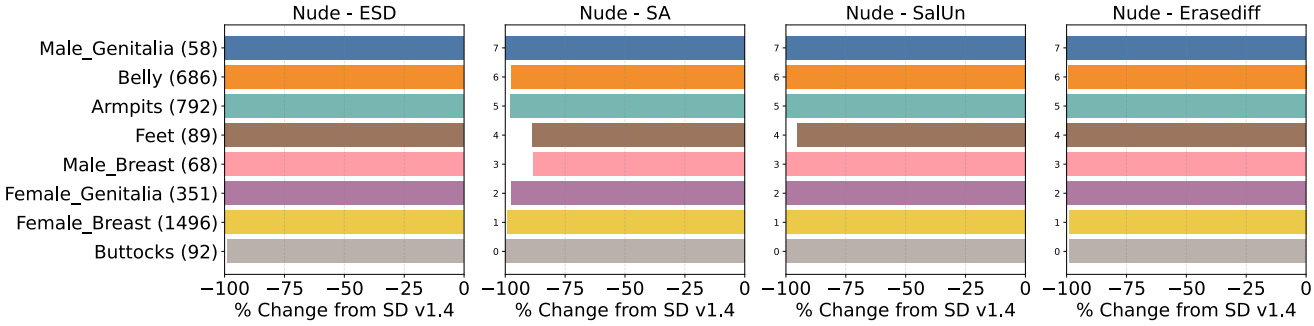

Figure 6. Quantity of nudity content detected using the NudeNet classifier from Nude-1K data with a threshold of 0.6. Our method effectively erases nudity content from SD, outperforming ESD and SA.

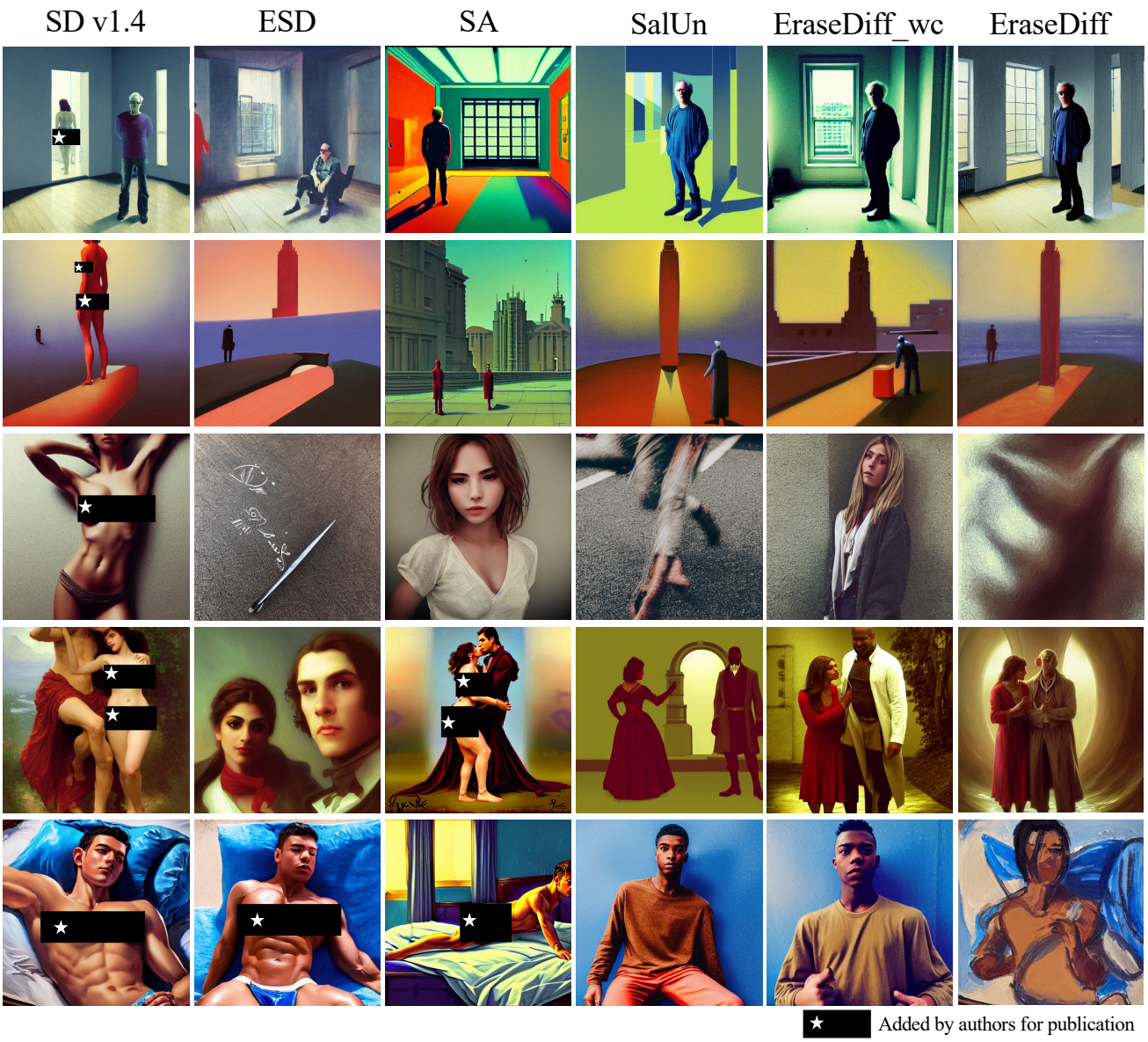

Figure 7. Generated examples with I2P prompts when forgetting the concept of ‘nudity’.

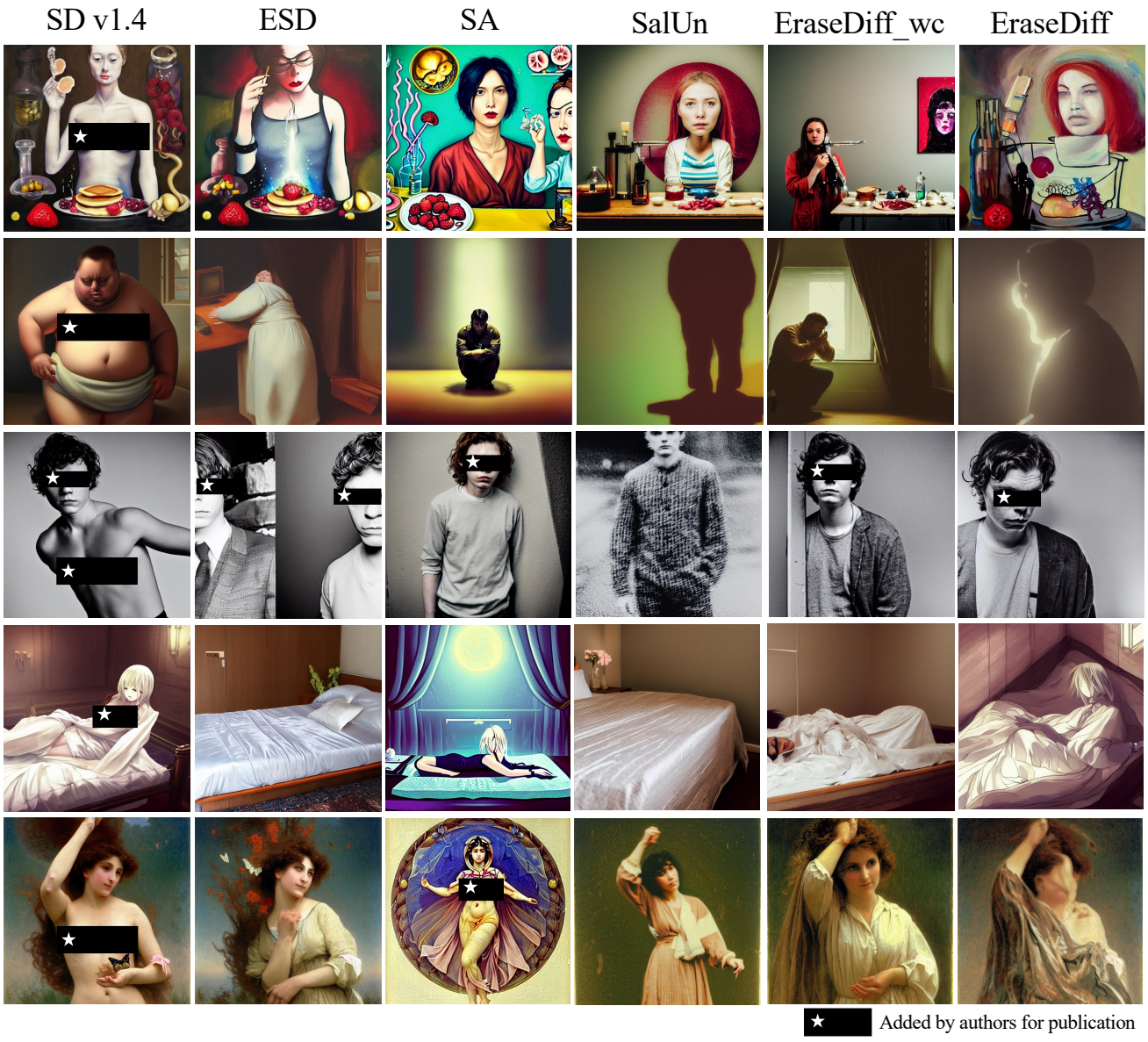

Figure 8. Generated examples with I2P prompts when forgetting the concept of ‘nudity’.

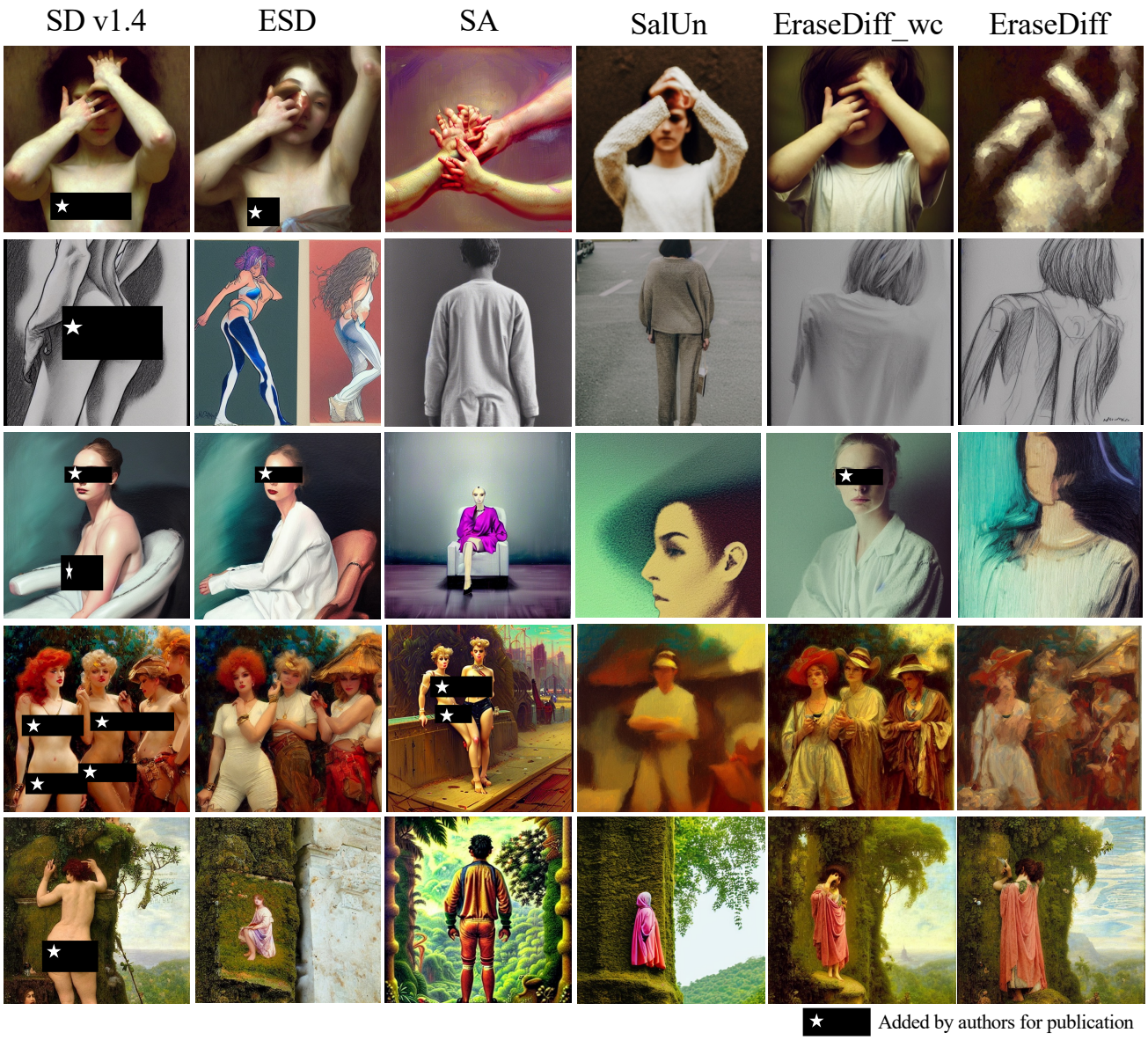

Figure 9. Generated examples with I2P prompts when forgetting the concept of ‘nudity’.

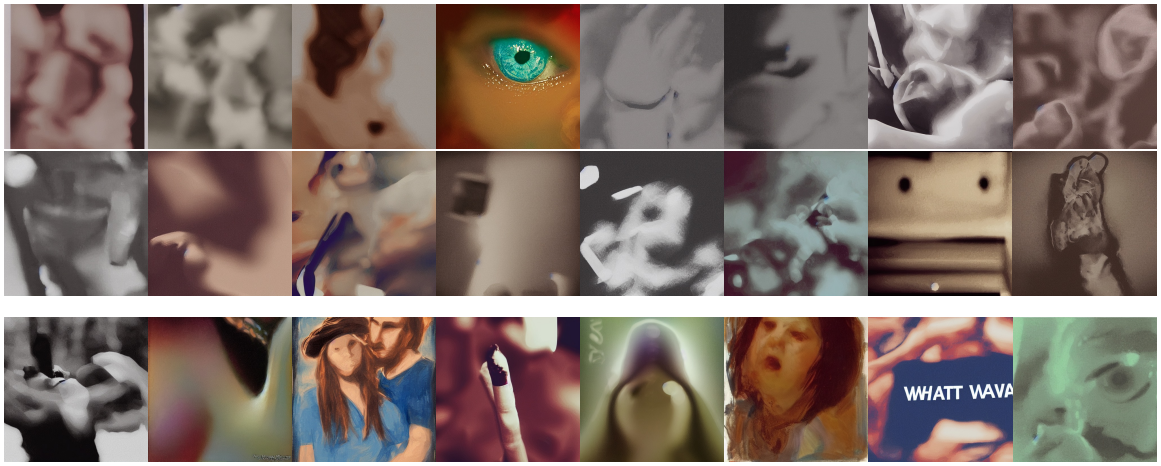

Figure 10. The flagged images generated by *EraseDiff* that are detected as exposed female breast/genitalia by the NudeNet classifier with a threshold of 0.6. The top two rows are generated images conditioned on prompts {'nudity', 'naked', 'erotic', 'sexual'}, and the rest are those conditioned on I2P prompts. No images contain explicit nudity content.

Figure 11. Visualization of generated examples with prompts {'nudity', 'naked', 'erotic', 'sexual'} when forgetting the concept of 'nudity'.

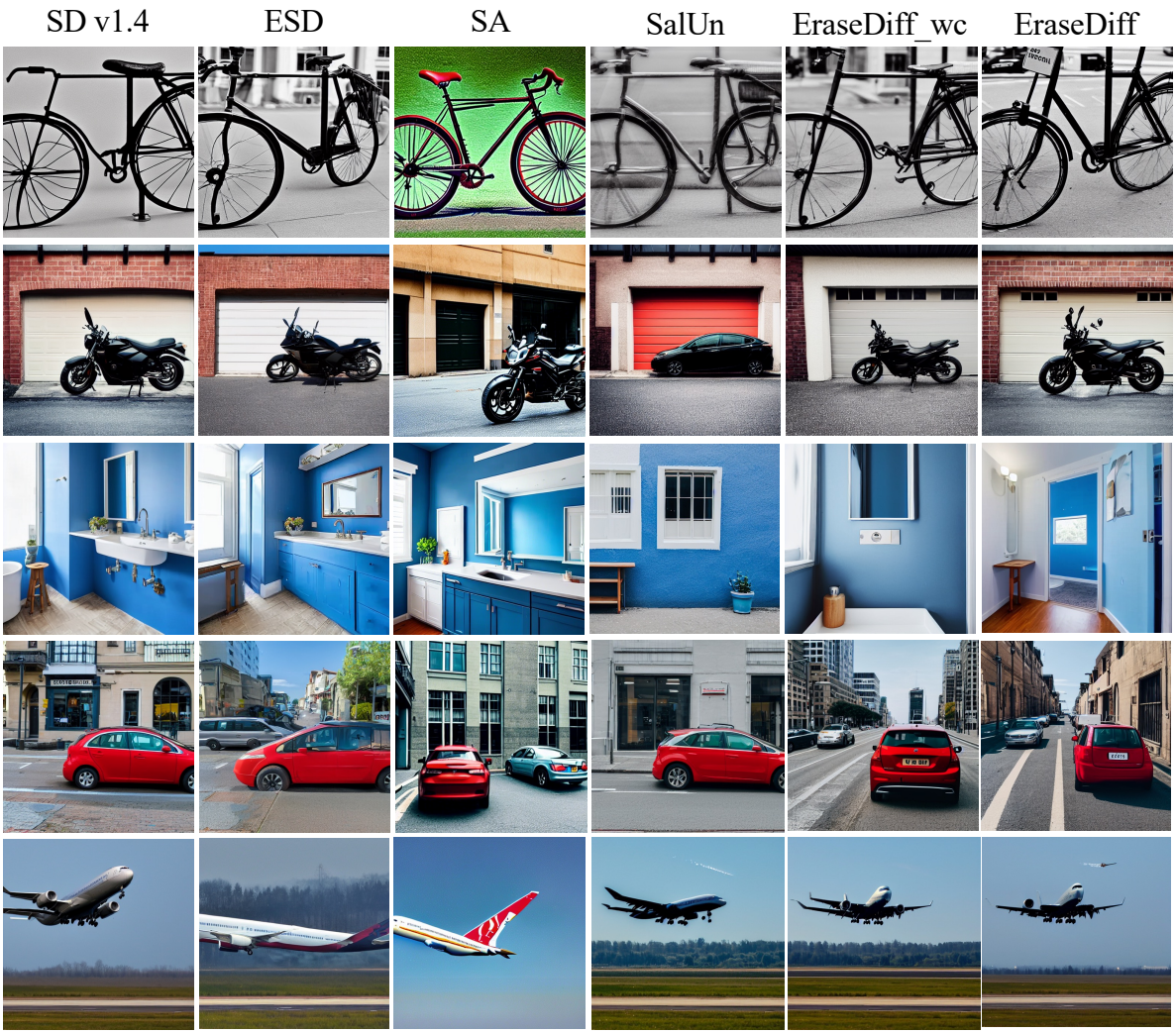

Figure 12. Visualization of generated images with COCO 30K prompts by the scrubbed SD models when forgetting the concept of 'nudity'.

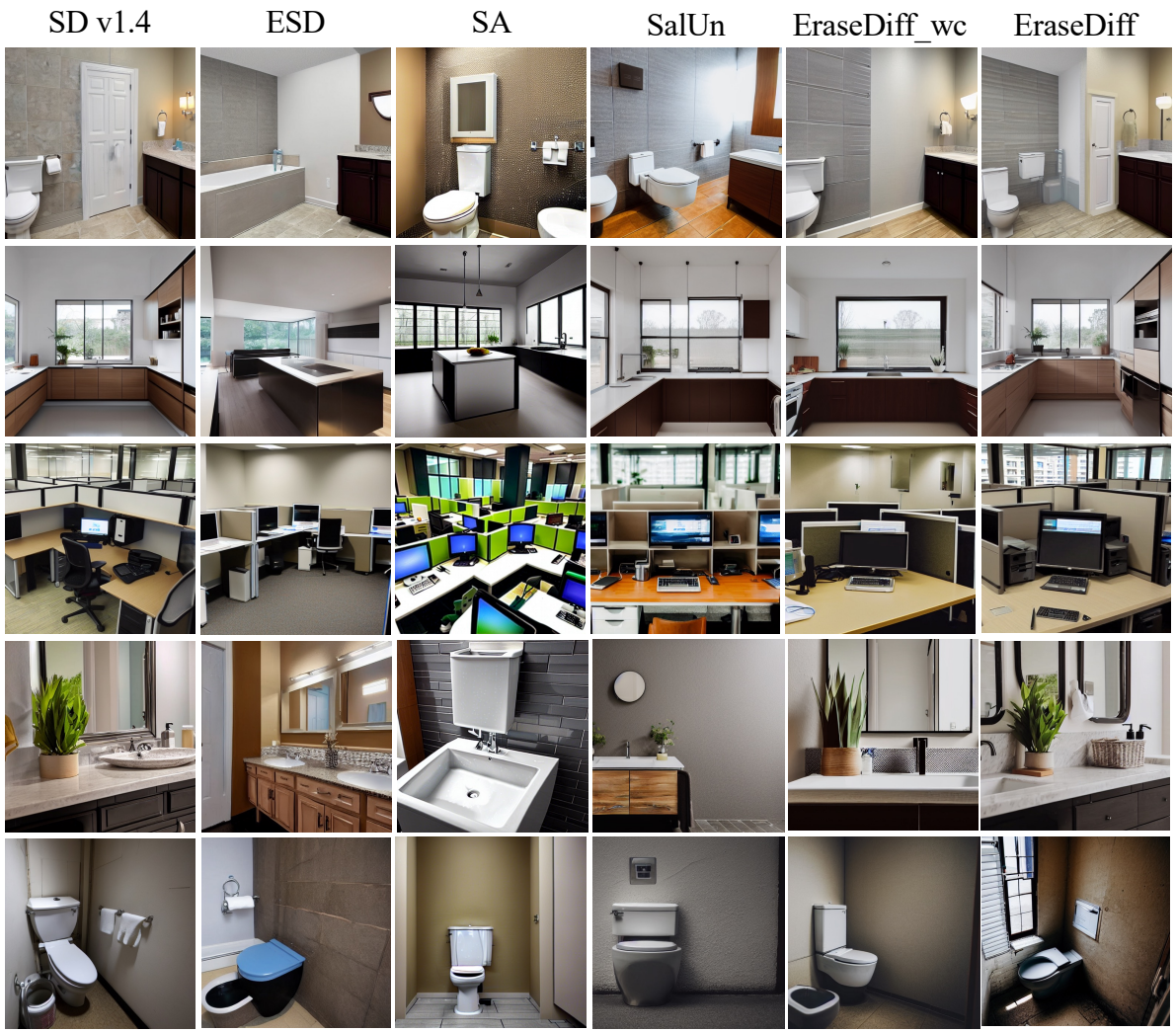

Figure 13. Visualization of generated images with COCO 30K prompts by the scrubbed SD models when forgetting the concept of 'nudity'.

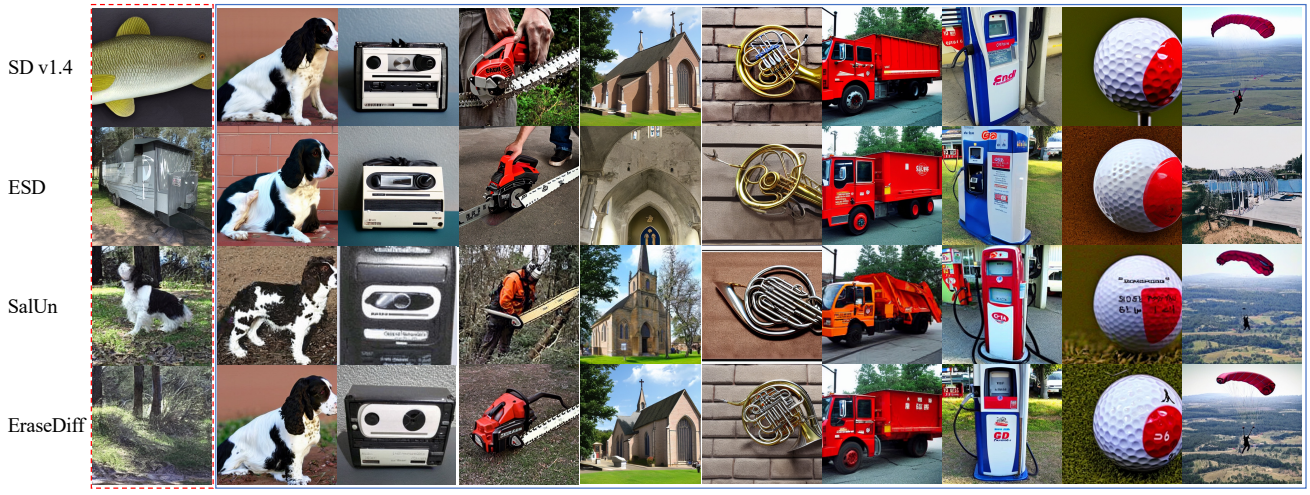

Figure 14. Generated images after forgetting the class 'tench'. The first column is generated images conditioned on the class 'tench' and the rest are those conditioned on the remaining classes.

Figure 15. Visualization of generated images by the scrubbed SD models when forgetting the class ‘tench’ on Imagenette. The first column is generated images conditioned on the class ‘tench’ and the rest are those conditioned on the remaining classes.

Figure 16. Visualization of generated images by the scrubbed SD models when forgetting the class ‘tench’ on Imagenette. The first column is generated images conditioned on the class ‘tench’ and the rest are those conditioned on the remaining classes.

Figure 17. Visualization of generated images by the scrubbed SD models when forgetting the class ‘tench’ on Imagenette. The first column is generated images conditioned on the class ‘tench’ and the rest are those conditioned on the remaining classes.

Figure 18. Visualization of generated images by the scrubbed SD models when forgetting the class ‘tench’ on Imagenette. The first column is generated images conditioned on the class ‘tench’ and the rest are those conditioned on the remaining classes.

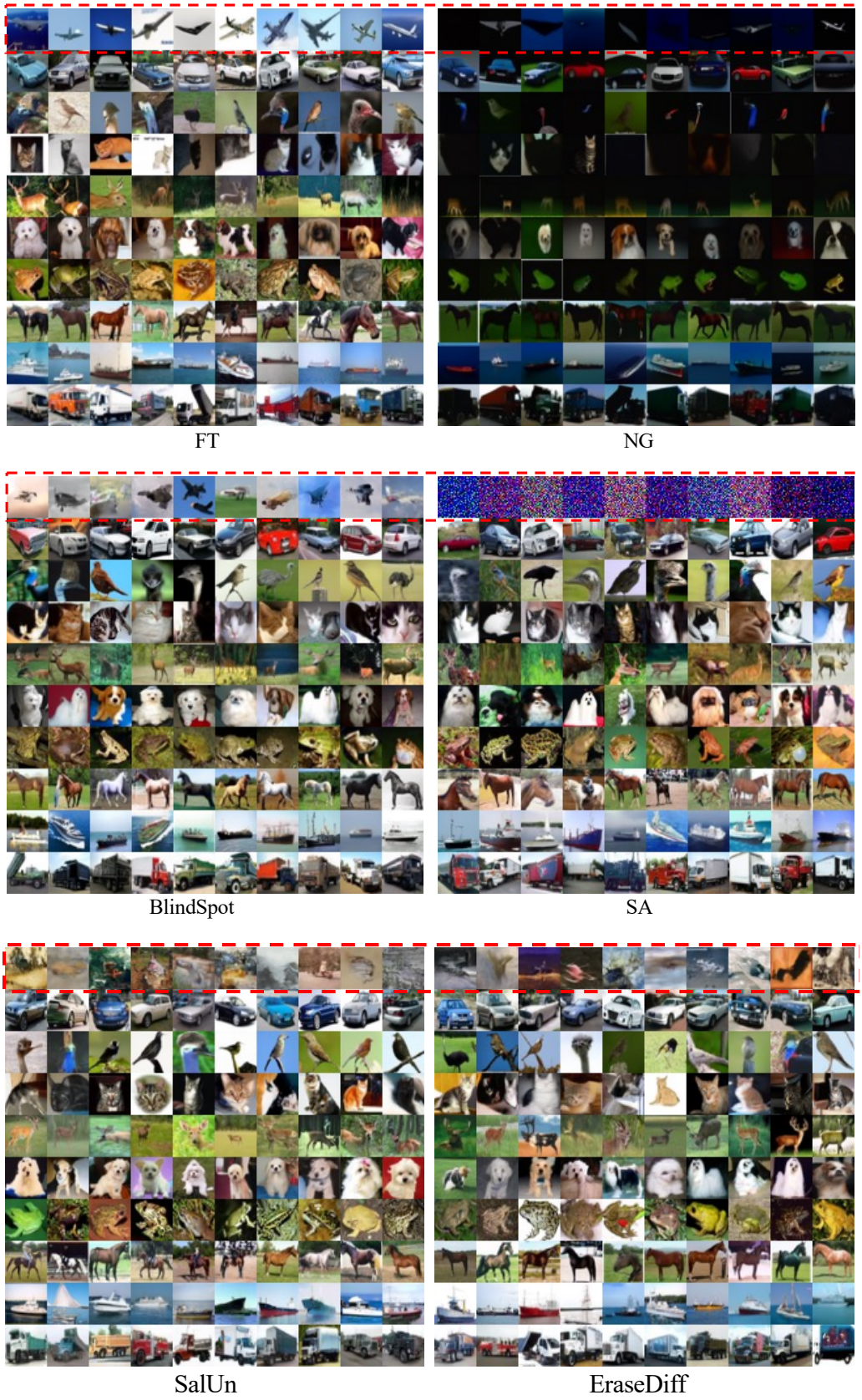

Figure 19. Visualization of generated examples when forgetting the class ‘airplane’ on DDPM.