R-ESC: Robustly Erasing Space Concepts via Stochastic Feature Remapping

Kang Eun Jeon[1,2], Yunsung Kang[3], Do Yeong Kang[3], Tae-Young Lee[4], Gyeong-Moon Park[4] ✉, and Jong Hwan Ko[3] ✉

[1] KAIST [2] DGIST [3] Sungkyunkwan University [4] Korea University
kejeon@kaist.ac.kr, {kangsung7,ksps7106}@skku.edu,
{tylee0415,gm-park}@korea.ac.kr, jhko@skku.edu
✉ Corresponding authors.

Abstract. Concept erasure and machine unlearning have emerged as a critical paradigm for enabling models to remove unwanted knowledge from learned representations at the level of classes, attributes, or higher-level semantics, motivated by the safety concerns surrounding modern generative AI models. However, existing approaches face three persistent challenges: (1) they often degrade utility on the remain concepts, (2) they often leave residual feature-level signals that enable recovery of the forget concepts via fine-tuning or representation probing, and (3) they impose substantial computational or memory overhead, limiting scalability. Prior work, such as Erasing Space Concept (ESC), moves toward structured removal in representation space, but achieving robust erasure with low overhead remains a challenge. To address this issue, we propose Robustly Erasing Space Concepts (R-ESC), a framework for robustly erasing space concepts via stochastic feature remapping. R-ESC introduces three components: (i) prototype-orthogonal projection that reduces interference by decorrelating forget and remain prototypes prior to erasure, (ii) stochastic multi-target remapping that blends forget prototypes into multiple remain prototypes to prevent re-separation and hinder recovery, and (iii) activation-mean prototypes that compress the erasure procedure to a single forward pass, yielding linear-time computation with constant additional memory. Across extensive experiments, R-ESC maintains remain data utility, strengthens resistance to feature-level recovery, and scales to large models and datasets, providing a practical pathway toward robust and efficient concept erasure.

1 Introduction

The rapid advancement of large-scale AI models has highlighted the critical need for model safety and alignment [1]. Ensuring that models do not inadvertently generate biased, inappropriate, or sensitive content [24, 27] has become a top priority for developers and regulators alike [21]. Enforcing such restrictions is particularly difficult in deep learning, as selectively ablating knowledge from a highly monolithic and polysemantic model [4] is a non-trivial task: the influence of specific concepts is deeply embedded and entangled across model parameters.

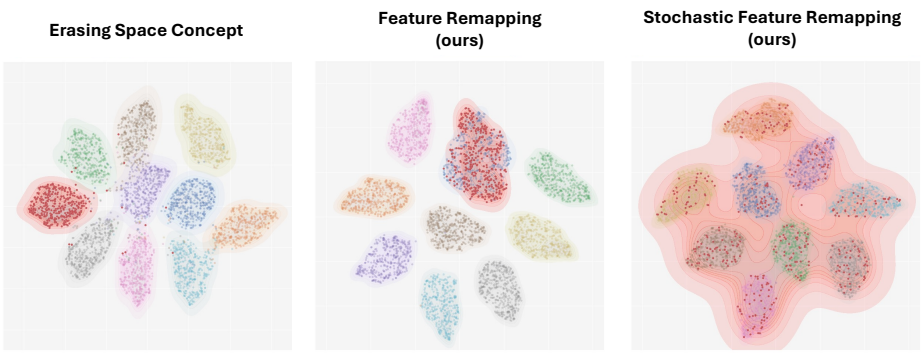

Fig. 1: t-SNE visualization of the latent feature space of different concept erasure methods on CIFAR-10 using AllCNN. Red points denote samples of the forget concept. ESC retains a separable red cluster after concept erasure, indicating residual feature-level information. In contrast, our remapping method absorbs the red cluster into the blue remain cluster, leaving the two concepts indistinguishable in the latent feature space. Stochastic remapping further disperses features, promoting robust erasure.

This challenge has sparked the emergence of the field of machine unlearning (MU) [2,20], more specifically concept erasure (CE) [6,10], which aims to remove the influence of designated forget concepts from trained models while preserving the model's performance or utility on benign or remain concepts.

Despite the recent advances in CE, current approaches fall short of achieving *complete knowledge deletion* [14, 19]. Prior studies have shown that traces of erased concepts can persist in latent feature space, enabling recovery of forgotten concepts and effectively *reversing* the unlearning [11]. To mitigate this, Erasing Space Concept (ESC) [19] performs *feature-level erasure* by identifying the principal components or *prototypes* that characterize the forget concepts via singular value decomposition (SVD). A projection layer is then inserted to erase the subspace spanning the forget concepts by removing the most influential k prototypes. To date, ESC achieves state-of-the-art (SOTA) performance in CE.

While ESC marks an important step toward stronger erasure guarantees, it shares critical limitations with other CE approaches. First, *utility degradation*: aggressively erasing feature subspaces often disrupts representations of remain concepts, causing significant performance drops. Second, *reversibility*: forget concepts often remain cohesive and separable in the latent space (see Fig. 1), leaving them highly vulnerable to recovery via light fine-tuning. Third, *inefficiency*: while ESC removes the substantial retraining overhead of traditional methods, it faces severe memory bottlenecks, as storing activations and performing SVD scales poorly with dataset size. These challenges highlight the need for a scalable framework that preserves utility while ensuring robust, irreversible feature-level erasure. Throughout, *robustness* refers specifically to *irreversibility*: erased knowledge cannot be recovered through subsequent fine-tuning or probing.

To address these gaps, we propose **R-ESC: Robustly Erasing Space Concepts**. Instead of leaving forget features as a distinct cluster in the latent space, R-ESC remaps them into the remain data distribution to disrupt their cohesive-separable structure, making the two indistinguishable at feature level (see Fig. 1). This robust feature-level erasure is enabled by three core innovations:

- **Prototype-orthogonal projection.** We decorrelate entangled forget and remain prototypes prior to erasure, ensuring only concept-specific information is targeted to preserve utility.
- **Remapping for robustness.** R-ESC scatters forget prototypes across multiple remain prototypes; destroying the separable cluster structure significantly impedes adversarial recovery via fine-tuning or probing.
- **Erasure efficiency.** Using concept-wise activation means yields linear time and constant space complexity, allowing R-ESC to scale effortlessly to large datasets and models.

Through these innovations, R-ESC addresses the long-standing limitations of typical erasure approaches by simultaneously preserving remain utility, ensuring robustness at the feature level, and enabling scalability.

2 Related Works

Knowledge Deletion Problem Formulation. A key limitation of CE/MU methods is their strong emphasis on output-centric evaluation metrics. For instance, MU methods often compare outputs to a retrain-from-scratch "gold standard" [2, 8], while CE methods rely on verifying the absence of forget concepts in generated outputs or classification logits [5,16]. This output-centric paradigm is fundamentally insufficient because it cannot measure whether the underlying knowledge has actually been erased from the latent feature space. As highlighted by recent studies on evaluation validity and concept recovery [7, 9, 17, 26], even when a method successfully scrubs forget concepts from the final output layer, it often leaves the underlying feature representations largely intact. By failing to decouple the forget concepts from correlated remain features, these residual representations act as a bridge that leaves the model highly vulnerable to adversarial recovery via representation probing or light fine-tuning [7].

To address these vulnerabilities, the *knowledge deletion* (KD) problem formulation [19] extends existing paradigms by demanding that forget knowledge be structurally disabled at the feature level. Crucially, it introduces rigorous fine-tuning evaluations to directly measure true erasure and ensure residual knowledge cannot be reconstructed. By prioritizing feature-level robustness alongside standard benchmarks, knowledge deletion provides a more thorough framework to evaluate model safety. We therefore base our work on this formulation to rigorously evaluate and achieve robust CE. Here, we use *robustness* in the sense of *irreversibility*—the inability to recover erased knowledge via fine-tuning or probing; we discuss this distinction further in Appendix I.

Erasing Space Concept (ESC). *Erasing Space Concept (ESC)* [19] is a pioneering approach to achieving feature-level erasure. The core idea is to remove the principal components, or prototypes, of the forget data from the latent feature space, thereby erasing knowledge at the feature level. Prototypes are representative vectors of the feature distribution, typically obtained through methods such as factorization or clustering. Concretely, the forget dataset is passed through the feature extractor $h_\psi(\cdot)$ of the original model to produce the feature matrix $\mathbf{Z}_f \in \mathbb{R}^{d \times N_f}$, where d is the feature dimensions and N_f is the number of forget data. This matrix is then decomposed via singular value decomposition (SVD), $\mathbf{Z}_f = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, where the columns of $\mathbf{U}$ represent the principal directions of the forget feature space. ESC constructs a projection layer that suppresses these directions by pruning the top-k directions. The resulting unlearned feature extractor and full ESC model are defined as

$$h_{\psi_p}(\mathbf{x}) = \mathbf{U}_p \mathbf{U}_p^\top h_\psi(\mathbf{x}), \quad f_{\text{ESC}}(\mathbf{x}) = g_\phi(h_{\psi_p}(\mathbf{x})), \tag{1}$$

where $\mathbf{U}_p = \mathbf{U}[k:]$ denotes the pruned subspace, and $g_\phi(\cdot)$ denotes the classification head. To further refine this process, ESC also proposed a training-based variant in which a learnable mask is applied over the left singular vectors (more details in Appendix §D.2). ESC has demonstrated state-of-the-art results across both conventional erasure benchmarks and the new Knowledge Retention (KR) metric, which measures feature-level erasure efficacy. Its efficiency is another strength: since it requires only a single forward pass and an SVD on the collected features, ESC is training-free and scalable to large models.

Limitations of ESC. Despite these advantages, ESC exhibits several limitations. First, it frequently leads to degraded utility on the remain dataset. This arises from the high correlation between the principal directions of forget and remain data, such that removing the former inevitably disrupts the latter (more details in §3.1). Second, even though ESC erases the forget subspace, the forget feature vectors often remain cohesive and separable in the latent space (see Fig. 1). This makes them vulnerable to recovery through light fine-tuning, raising concerns about privacy leakage. Third, although compute-efficient, ESC is not memory-efficient: storing all activations and performing SVD requires memory proportional to the number of forget samples and feature dimensions (i.e., $O(N_f d)$), which can cause out-of-memory failures on large-scale datasets. These challenges highlight the open problems that remain in achieving utility-preserving, robust, and scalable erasure.

3 Methods

To address the limitations of existing subspace erasure methods like ESC, we propose **Robustly Erasing Space Concepts (R-ESC)**; an overview of our framework is illustrated in Fig. 2. Moving beyond simple subspace erasure, R-ESC achieves robustness by warping the latent feature space, effectively remapping the underlying subspace defined by forget prototypes into those of the remain concepts.

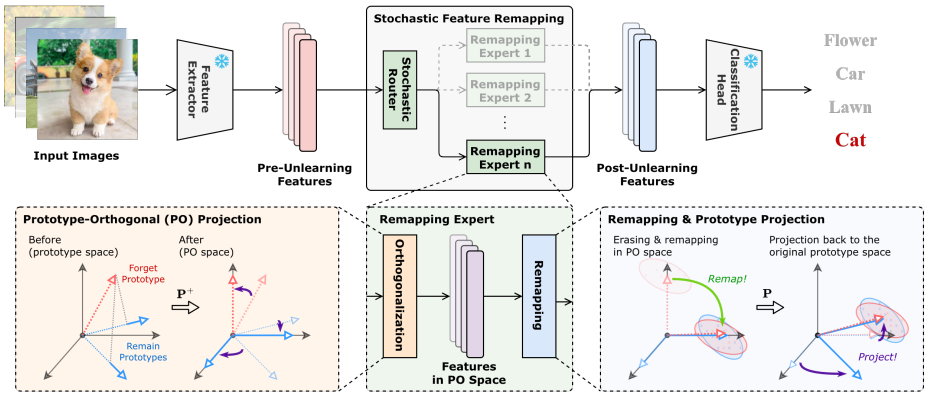

Fig. 2: An overview of our proposed R-ESC framework.

The R-ESC layer executes this structural projection through a router and a set of remapping experts. For a given pre-unlearning feature, the router randomly assigns an expert to apply two sequential operations: (i) projection into a disentangled *prototype-orthogonal (PO) space*, and (ii) erasing and remapping of the forget basis vectors. If a single expert is used to fold the forget subspace into a single remain prototype, it disrupts linear separability. However, by extending this architecture to multiple experts, the warped space scatters incoming forget features across several different remain prototypes. This breaks the cohesive structure of the forgotten knowledge, leaving little residual structure for linear probes to exploit while safely preserving model utility.

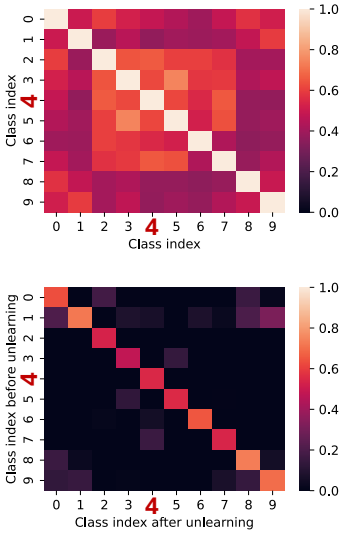

Fig. 3: The *top* plot displays cosine similarities between class prototypes before erasure, highlighting the initial entanglement among concepts. The *bottom* plot compares similarities between original and post-erasure prototypes, illustrating the collateral damage inflicted on remain concepts. Prototypes are extracted as class-wise activation means. Class 4 corresponds to the forget class.

3.1 Prototype-Orthogonal Projection

The first step of R-ESC is Prototype-Orthogonal (PO) projection, which is essential for enabling stable erasure and remapping. Our key observation is that forget and remain prototypes are often highly entangled in the latent feature space. Consequently, erasing or editing forget prototypes inadvertently distorts remain prototypes, degrading model utility and yielding incomplete erasure.

Empirical evidence of this entanglement is shown in Fig. 3, where we present two sets of cosine similarity measurements on CIFAR-10. First, Fig. 3 (*top*) computes the inter-class similarity between original prototypes; on average, the similarity between forget and remain prototypes is approximately 0.5, peaking at 0.77. This confirms that forget and remain representations are not naturally orthogonal but are instead substantially entangled in the feature space.

Second, to examine the consequence of this entanglement during erasure, Fig. 3 (*bottom*) measures the similarity between the original prototypes and the post-erasure prototypes. Ideally, the similarity for remain concepts should stay near 1.0 (indicating preserved representations), while dropping to zero for the forget concept. In practice, however, we observe a sharp decrease across the board: the remain concepts drop from 1.0 to 0.52, while the forget concept falls only to 0.46—a level indistinguishable from the other remain concepts. This demonstrates that erasing a forget prototype without accounting for entanglement creates significant collateral damage, disrupting the entire representation space.

To mitigate this, we project input features into a *prototype-orthogonal space*, where the projected prototype vectors form an orthogonal basis. In this space, each prototype corresponds to an independent coordinate axis, ensuring that manipulating a forget prototype does not influence the others.[1] This can be achieved using the Moore-Penrose pseudoinverse of the prototype matrix $\mathbf{P} = [\mathbf{p}_1, \ldots, \mathbf{p}_k] \in \mathbb{R}^{d \times k}$ whose columns are class prototypes. The pseudoinverse $\mathbf{P}^+$ is computed as:

$$\mathbf{P}^+ = \mathbf{V}\,\mathbf{\Sigma}^{-1}\,\mathbf{U}^\top, \tag{2}$$

where $\mathbf{P} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ is the singular value decomposition (SVD) with singular vectors $\mathbf{U} \in \mathbb{R}^{d \times k}, \mathbf{V} \in \mathbb{R}^{k \times k}$ and diagonal singular value matrix $\mathbf{\Sigma} \in \mathbb{R}^{k \times k}$. By definition, $\mathbf{P}^+\mathbf{P} = \mathbf{I}_k$, ensuring that projected features $\mathbf{P}^+\mathbf{z}$ inhabit a PO space where each dimension corresponds to an independent coordinate for a specific concept prototype. By isolating concepts onto these orthogonal axes, subsequent erasing or remapping of forget prototypes can proceed with minimal distortion to remain prototypes, maximizing erasure effectiveness while preserving model utility.

3.2 Erasing and Remapping

Once features are projected into the PO space (i.e., $\mathbf{P}^+\mathbf{z}$), we can surgically erase forget concepts by suppressing their corresponding dimensions. Since each dimension of the PO space is isolated, zeroing out a forget coordinate removes its influence without affecting the others.

Formally, let $\mathbf{s} \in \{0, 1\}^k$ be a binary selector vector where $s_i = 1$ if the i-th prototype belongs to the forget set, and $s_i = 0$ otherwise. Then the unlearned

[1] Strictly speaking, only orthogonality between forget and remain prototypes is required. Enforcing this selectively, however, is mathematically complex, so we adopt full mutual orthogonality for brevity.

feature, $\hat{\mathbf{z}}$, is given by:

$$\hat{\mathbf{z}} = \mathbf{P}\big(\mathbf{P}^+ - \mathrm{diag}(\mathbf{s})\mathbf{P}^+\big)\mathbf{z}, \tag{3}$$

where $\mathrm{diag}(\mathbf{s})$ is a diagonal matrix constructed from the selector vector $\mathbf{s}$.

One limitation of this formulation is that $\mathbf{P}\mathbf{P}^+$ projects $\mathbf{z}$ entirely into the low-rank prototype span, discarding non-concept information that resides outside this subspace. To preserve this information, we add a *complement-space projection* term, akin to a residual connection. The full unlearned feature $\hat{\mathbf{z}}$ is then given by:

$$\hat{\mathbf{z}} = \underbrace{\mathbf{P}\big(\mathbf{P}^+ - \mathrm{diag}(\mathbf{s})\mathbf{P}^+\big)\mathbf{z}}_{\text{erased PO space}} + \underbrace{\big(\mathbf{I} - \mathbf{P}\mathbf{P}^+\big)\mathbf{z}}_{\text{complement space}} . \tag{4}$$

This ensures a full-rank transformation: the influence of forget prototypes is removed, while all information outside the prototype span remains intact.

This expression can be elegantly simplified to:

$$\hat{\mathbf{z}} = (\mathbf{I} - \mathbf{P}_f\mathbf{P}^+)\mathbf{z}, \tag{5}$$

where $\mathbf{P}_f = \mathbf{P}\,\mathrm{diag}(\mathbf{s})$ is a matrix containing only the forget prototypes (with remain prototypes zeroed out). Notice that $\mathbf{P}_f\mathbf{P}^+$ acts as a linear operator that surgically extracts the components of $\mathbf{z}$ aligned with the forget prototypes. Subtracting this term ensures that $\hat{\mathbf{z}}$ lies entirely in the subspace orthogonal to the forget concepts, thereby erasing their influence.

To extend erasure into remapping, we warp the representation space by redirecting the energy of forget prototypes toward remain prototypes. We substitute the simple suppression with an active spatial shift:

$$\hat{\mathbf{z}} = \big(\underbrace{\mathbf{I}_d - \mathbf{P}_f\mathbf{P}^+}_{\text{erasure}} + \underbrace{\mathbf{P}_m\,\mathrm{diag}(\mathbf{s})\mathbf{P}^+}_{\text{remapping}}\big)\mathbf{z}, \tag{6}$$

where $\mathbf{P}_m$ is a *remapping matrix* whose i-th column is set to a designated remain prototype $\mathbf{p}_j$ if the i-th prototype is a forget concept ($s_i = 1$), and zero otherwise. The first term performs the orthogonal erasure as before, while the second acts as the remapping mechanism: whenever a forget prototype is activated in the PO space, it is remapped to the designated remain prototype. In effect, this process merges the latent representations of forget data into the remain subspace, making them indistinguishable and preventing recovery via linear probing.

3.3 Stochastic Remapping with Multiple Experts

While redirecting each forget prototype to a single remain prototype effectively blurs class separability, the remapped forget features preserve a cohesive cluster structure. As previously noted, this residual cohesion can be exploited by carefully tuned linear probes to recover the erased concepts. To systematically break this residual structure, we introduce *stochastic feature remapping*. Specifically,

we extend our remapping mechanism by introducing multiple remapping layers, each mapping a given forget prototype to different remain prototypes. This approach draws inspiration from the Mixture of Experts (MoE) architecture. However, rather than functioning to improve prediction accuracy, our 'experts' exist solely to scatter forget features across distinct remain clusters.

To allocate inputs across these experts, we employ an input-independent *stochastic router* that assigns each input $\mathbf{z}$ uniformly at random during the forward pass. This stochasticity guarantees that the features of a single forget concept are widely scattered across multiple remain clusters, maximizing the geometrical disruption of prior cohesive structures. Furthermore, because the stochastic router requires zero training, ensures an even workload distribution, and introduces negligible computational overhead during inference, it scales seamlessly to large models and datasets. We adopt this stochastic routing mechanism as the default formulation for R-ESC. The complete pseudo-code summarizing our erasure and stochastic remapping framework is provided in Appendix G (Algorithm C).

While we also explored a trained conditional router, we encountered challenges with expert specialization collapse that reduced scattering diversity and hindered robust erasure. Detailed analysis and formulations of this conditional approach are deferred to Appendix §G.

3.4 Erasure Efficiency

R-ESC constructs prototype vectors by simply *averaging concept-wise activations*. Because this approach avoids expensive clustering or matrix factorization, and all subsequent erasure operations involve only lightweight linear algebra, our framework is entirely training-free. This design requires only $O(Nd)$ computational complexity to assemble prototypes from N samples, and $O(dk)$ memory to store k concept prototypes. Consequently, R-ESC scales effortlessly to large models and datasets with minimal computational overhead.

4 Experiments

4.1 Experimental Setup & Metrics

Experiment Setup & Overview. We follow the standard experimental setup commonly adopted in the CE literature to ensure a fair comparison across a wide set of classical and recent baselines (see Appendix §D and §D.2 for full baseline descriptions). All experiments are run on A100 and RTX 3090 GPUs.

For **image classification** tasks, we evaluate R-ESC on CIFAR-10 [15] using All-CNN [25], CIFAR-100 [15] using ResNet-18 [12], Tiny-ImageNet [18], and ImageNet [23] using ViT-base-16 [3]. In these experiments, we follow the standard class-wise erasure setting where 10% of the existing classes are targeted for removal. As reference, we include **Original**, a model trained on the full dataset $\mathcal{D}$, and **Retrain**, a model trained only on the remain data $\mathcal{D}_r$.

Table 1: Accuracy and KR performance evaluated using CIFAR-10 with AllCNN, CIFAR-100 with ResNet-18, and Tiny-ImageNet with ViT. The table presents the mean and standard deviation (mean ± std) across three trials with the best value highlighted in bold.

CIFAR-10

Method	D_f (↓)	D_r (↑)	D_ft (↓)	D_rt (↑)	HM (↑)	HM_t (↑)
Original	99.92	99.94	91.80	91.07	0.16	15.05
Retrain	0.00	99.15	0.00	92.04	99.57	95.86
Finetune	0.00 ± 0.00	90.86 ± 2.66	0.00 ± 0.00	84.89 ± 2.39	95.20 ± 0.80	91.83 ± 0.82
NG	6.35 ± 0.08	89.62 ± 0.22	5.72 ± 0.11	81.26 ± 0.17	91.59 ± 0.03	87.28 ± 0.03
RL	0.00 ± 0.00	97.52 ± 0.29	0.00 ± 0.00	90.14 ± 0.06	98.74 ± 0.08	94.82 ± 0.02
BS	9.87 ± 0.02	95.23 ± 0.01	1.00 ± 0.01	85.75 ± 0.01	92.61 ± 0.01	87.83 ± 0.01
Lau	0.18 ± 0.00	86.57 ± 5.67	0.13 ± 0.01	79.51 ± 3.79	92.71 ± 1.84	88.53 ± 1.39
ESC	11.22 ± 1.18	88.84 ± 0.40	10.70 ± 0.87	81.27 ± 0.50	88.81 ± 0.74	85.10 ± 0.65
ESC-T	0.03 ± 0.02	88.88 ± 0.22	0.00 ± 0.00	81.21 ± 0.19	94.10 ± 0.12	89.63 ± 0.11
Remap	0.00 ± 0.00	99.87 ± 0.00	0.00 ± 0.00	91.16 ± 0.01	**99.94 ± 0.00**	**95.38 ± 0.01**
R-ESC	0.00 ± 0.00	99.87 ± 0.01	0.00 ± 0.00	91.02 ± 0.11	99.93 ± 0.01	95.30 ± 0.06

CIFAR-10 (KR setting; lr = 0.1)

Method	D_f (↓)	D_r (↑)	D_ft (↓)	D_rt (↑)	HM (↑)	HM_t (↑)
Original	99.88	99.95	91.20	91.07	0.24	16.05
Retrain	72.62	97.06	72.90	88.01	42.71	41.44
Finetune	79.24 ± 11.93	93.80 ± 0.14	76.91 ± 3.90	87.30 ± 0.19	33.45 ± 21.55	36.36 ± 6.35
NG	94.34 ± 0.09	98.05 ± 0.01	83.97 ± 0.04	88.80 ± 0.00	10.69 ± 0.28	27.16 ± 0.07
RL	87.17 ± 1.69	97.53 ± 0.00	81.19 ± 0.83	89.38 ± 0.02	22.55 ± 4.07	31.03 ± 1.53
BS	96.43 ± 0.01	98.90 ± 0.00	85.50 ± 0.05	89.36 ± 0.00	6.88 ± 0.02	24.95 ± 0.10
Lau	90.60 ± 0.31	94.29 ± 0.10	80.70 ± 0.32	85.37 ± 0.03	17.06 ± 0.83	31.47 ± 0.54
ESC	82.97 ± 0.23	93.45 ± 0.21	73.60 ± 0.17	84.98 ± 0.16	28.80 ± 0.34	40.28 ± 0.19
ESC-T	87.34 ± 1.11	94.14 ± 0.35	78.80 ± 1.21	85.24 ± 0.31	22.31 ± 1.75	33.94 ± 1.58
Remap	33.20 ± 57.50	96.17 ± 6.40	29.83 ± 51.67	87.76 ± 5.92	66.89 ± 57.24	69.78 ± 44.34
R-ESC	10.79 ± 9.85	95.96 ± 3.36	12.87 ± 8.96	86.82 ± 3.30	**92.37 ± 6.91**	**86.91 ± 6.17**

CIFAR-100

Method	D_f (↓)	D_r (↑)	D_ft (↓)	D_rt (↑)	HM (↑)	HM_t (↑)
Original	100.00	99.98	67.00	80.81	0.00	46.86
Retrain	0.00	96.64	0.00	72.00	98.29	83.72
Finetune	0.00 ± 0.00	87.41 ± 0.18	0.00 ± 0.00	69.39 ± 0.25	93.28 ± 0.06	81.93 ± 0.12
NG	29.19 ± 0.32	96.79 ± 0.02	5.65 ± 0.22	70.06 ± 0.09	81.79 ± 0.18	80.41 ± 0.04
RL	1.27 ± 1.34	82.19 ± 0.27	1.00 ± 2.00	67.58 ± 0.04	89.70 ± 0.03	80.33 ± 0.16
BS	1.93 ± 0.54	58.57 ± 10.75	1.00 ± 2.00	41.48 ± 4.33	73.34 ± 6.86	58.45 ± 5.39
Lau	2.60 ± 0.00	96.35 ± 0.01	0.00 ± 0.00	70.08 ± 0.01	96.87 ± 0.00	82.41 ± 0.01
ESC	0.50 ± 0.38	88.85 ± 0.32	0.00 ± 0.00	64.72 ± 0.17	93.77 ± 0.05	78.58 ± 0.10
ESC-T	0.00 ± 0.00	99.03 ± 0.00	0.00 ± 0.00	73.88 ± 0.00	99.51 ± 0.00	84.98 ± 0.00
Remap	0.00 ± 0.00	99.98 ± 0.00	0.00 ± 0.00	80.82 ± 0.00	**99.99 ± 0.00**	**89.39 ± 0.00**
R-ESC	0.00 ± 0.00	99.98 ± 0.00	0.00 ± 0.00	80.22 ± 0.02	**99.99 ± 0.00**	89.03 ± 0.02

CIFAR-100 (KR setting; lr = 0.1)

Method	D_f (↓)	D_r (↑)	D_ft (↓)	D_rt (↑)	HM (↑)	HM_t (↑)
Original	100.00	99.98	67.00	80.41	0.00	46.80
Retrain	57.20	97.20	58.00	71.66	59.43	52.96
Finetune	66.31 ± 2.46	92.88 ± 0.03	53.66 ± 0.89	72.70 ± 0.10	49.37 ± 2.89	56.59 ± 0.41
NG	96.80 ± 0.03	99.65 ± 0.00	52.99 ± 0.67	74.94 ± 0.01	6.19 ± 0.09	57.76 ± 0.36
RL	70.59 ± 1.39	88.98 ± 0.12	56.85 ± 16.67	71.82 ± 0.13	44.16 ± 1.90	53.58 ± 9.79
BS	61.07 ± 15.71	86.49 ± 1.55	33.91 ± 6.00	60.65 ± 0.39	53.31 ± 11.80	63.18 ± 1.89
Lau	94.53 ± 0.04	99.39 ± 0.00	48.98 ± 2.00	74.18 ± 0.00	10.36 ± 0.12	60.42 ± 1.01
ESC	99.60 ± 0.00	99.92 ± 0.00	59.65 ± 1.56	75.90 ± 0.01	0.80 ± 0.00	52.65 ± 1.15
ESC-T	96.07 ± 0.06	99.89 ± 0.00	61.99 ± 0.67	75.35 ± 0.01	7.55 ± 0.21	50.51 ± 0.49
Remap	51.88 ± 15.94	94.23 ± 1.88	41.63 ± 10.56	76.24 ± 1.61	62.76 ± 14.71	65.84 ± 7.35
R-ESC	0.07 ± 0.09	99.86 ± 0.18	1.27 ± 0.72	77.01 ± 0.06	**99.89 ± 0.14**	**86.53 ± 0.31**

Tiny-ImageNet

Method	D_f (↓)	D_r (↑)	D_ft (↓)	D_rt (↑)	HM (↑)	HM_t (↑)
Original	98.20	96.45	96.00	90.29	3.53	7.66
Retrain	0.00	99.98	0.00	85.23	99.99	92.03
Finetune	0.00 ± 0.00	66.89 ± 18.98	0.00 ± 0.00	55.49 ± 6.69	80.12 ± 9.42	71.35 ± 4.48
NG	0.01 ± 0.00	0.62 ± 0.00	0.00 ± 0.00	0.62 ± 0.00	1.24 ± 0.00	1.23 ± 0.00
RL	10.83 ± 4.26	96.98 ± 0.05	9.19 ± 5.45	85.14 ± 0.10	92.79 ± 1.17	87.73 ± 1.26
BS	36.07 ± 0.29	48.02 ± 0.19	35.43 ± 0.08	46.65 ± 0.16	54.84 ± 0.03	54.16 ± 0.05
Lau	0.00 ± 0.00	95.55 ± 0.00	0.00 ± 0.00	89.73 ± 0.00	97.73 ± 0.00	94.59 ± 0.00
ESC	0.10 ± 0.02	96.36 ± 0.05	0.03 ± 0.06	90.57 ± 0.05	98.10 ± 0.02	95.03 ± 0.05
ESC-T	0.00 ± 0.00	96.44 ± 0.03	0.00 ± 0.00	90.57 ± 0.04	**98.19 ± 0.02**	**95.05 ± 0.02**
Remap	0.03 ± 0.01	96.18 ± 0.01	0.00 ± 0.00	90.13 ± 0.03	98.04 ± 0.01	94.81 ± 0.02
R-ESC	0.02 ± 0.01	96.16 ± 0.02	0.00 ± 0.00	90.01 ± 0.09	98.03 ± 0.01	94.74 ± 0.05

Tiny-ImageNet (KR setting; lr = 0.1)

Method	D_f (↓)	D_r (↑)	D_ft (↓)	D_rt (↑)	HM (↑)	HM_t (↑)
Original	98.20	96.83	96.00	90.10	3.53	7.66
Retrain	78.57	99.87	76.00	80.70	35.29	37.00
Finetune	53.69 ± 5.46	71.19 ± 13.19	51.67 ± 10.00	59.63 ± 4.77	55.97 ± 0.44	53.20 ± 1.33
NG	17.11 ± 0.11	19.84 ± 0.33	16.10 ± 0.03	18.62 ± 0.14	32.02 ± 0.53	30.48 ± 0.24
RL	91.21 ± 0.25	97.49 ± 0.04	84.90 ± 0.01	85.30 ± 0.24	16.10 ± 0.71	25.66 ± 0.02
BS	72.25 ± 0.02	74.26 ± 0.03	68.83 ± 0.12	69.97 ± 0.02	40.41 ± 0.02	43.12 ± 0.10
Lau	96.69 ± 0.00	96.86 ± 0.00	89.70 ± 0.00	90.18 ± 0.00	6.40 ± 0.00	18.49 ± 0.00
ESC	15.78 ± 0.84	96.88 ± 0.02	13.67 ± 1.06	90.54 ± 0.08	90.10 ± 0.49	88.38 ± 0.52
ESC-T	95.47 ± 0.12	96.78 ± 0.03	89.57 ± 0.21	90.23 ± 0.11	8.66 ± 0.21	18.70 ± 0.33
Remap	55.18 ± 15.74	93.57 ± 1.65	51.03 ± 14.00	87.00 ± 1.46	59.60 ± 14.98	61.96 ± 11.95
R-ESC	0.50 ± 0.07	96.52 ± 0.01	0.43 ± 0.25	89.94 ± 0.07	**97.99 ± 0.04**	**94.51 ± 0.14**

For **text-to-image generation** tasks, we demonstrate concept-level erasure using Stable Diffusion v1.4 [22]. Following prior works [6,10], we apply prototype orthogonalization, erasure, and remapping to the cross-attention layers. Our experimental setup strictly follows the standard practice established by the SOTA diffusion erasure methods [5, 6, 10, 27], ensuring a fair comparison. We evaluate R-ESC on two distinct erasure tasks: artistic style erasure, where we remove the distinctive styles of both classical and modern artists [5]; and object erasure, utilizing prompts from Imagenette [13], a 10-class subset of ImageNet, as commonly evaluated in prior work [6].

4.2 Concept Erasure on Image Classification Models

Evaluations Metrics. We evaluate erasure performance by partitioning the dataset into remain training, forget training, remain test, and forget test data,

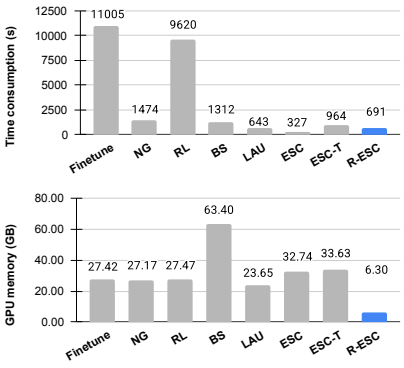

Fig. 4: Comparison of time (*top*) and GPU memory consumption (*bottom*).

Table 2: Ablation table to demonstrate the effectiveness of the proposed techniques.

CIFAR-10							
Method	**PO**	**D_f**	**D_r**	**D_ft**	**D_rt**	**HM**	**HM_t**
Erase	X	14.38	99.90	13.47	91.24	92.21	88.82
Remap	X	0.00	89.52	0.00	79.64	94.47	88.67
Erase	O	0.00	99.94	0.00	91.67	**99.97**	**95.65**
Remap	O	0.00	99.87	0.00	91.16	99.94	95.38
R-ESC	O	0.00	99.82	0.00	90.90	99.91	95.23
CIFAR-10 (KR setting; lr = 0.1)							
Method	**PO**	**D_f**	**D_r**	**D_ft**	**D_rt**	**HM**	**HM_t**
Erase	X	99.02	99.92	88.27	91.07	1.94	20.79
Remap	X	99.87	99.94	91.27	90.97	0.27	15.94
Erase	O	99.01	99.92	88.27	91.07	1.97	20.79
Remap	O	33.20	96.17	29.83	87.76	66.89	69.78
R-ESC	O	9.01	97.43	8.93	88.43	**93.94**	**89.61**

and measuring classification accuracy on each of them. The corresponding classification accuracies are denoted by D_r, D_f, D_rt, and D_ft, respectively. For each subset, we also report the **Knowledge Retention (KR)** metric to measure feature-level erasure performance (details in §D.3). Following prior work, we also employ the **Harmonic Mean (HM)** to measure balanced utility (details in §D.3). **Model Utility Evaluation.** Table 1 reports comprehensive results for the CE task across multiple dataset–model pairs, comparing our method against recent MU baselines. Due to space constraints, we present the most competitive recent methods here, while full results—including all baselines, additional learning rate configurations, and ImageNet results—are provided in the Appendix §E.1. Here, we compare two variants of our method: **Remap**, a single-expert model facilitating one-to-one remapping, and **R-ESC**, a multi-expert extension that enables one-to-many remapping for stronger erasure. Our proposed methods consistently achieve the best performance across all settings (see HM and HM_t). Notably, it surpasses training-based erasure methods that require orders of magnitude more compute. The gains are especially pronounced on CIFAR-10 and CIFAR-100, where R-ESC not only maintains remain-set accuracy (D_r and D_rt) but in some cases slightly improves it. These results underscore the effectiveness of our method enabling both near-perfect forgetting of unwanted knowledge and preservation of remain knowledge (see HM and HM_t).

KR Evaluation. The impact of R-ESC is even more striking under the KR evaluation. As shown in Table 1, existing baselines suffer from substantial recovery of forget accuracy (D_f and D_ft), approaching pre-erasure performance. In contrast, our method keeps forget accuracy down to the level of random guessing, decisively outperforming all baselines and even the retrain model. This demonstrates that R-ESC can achieve stronger erasure objectives beyond the conventional gold standard, establishing a new utility-focused and robust erasure paradigm.

Erasure Efficiency Evaluation. Beyond erasure effectiveness, R-ESC is also highly efficient and scalable. On CIFAR-10 and CIFAR-100, R-ESC performs

Fig. 5: Qualitative results of artistic style erasure of Van Gogh. The prompt: "*A depiction of a starry night over a quiet town, reminiscent of Van Gogh's famous painting*".

complete erasure in under 10 seconds while consuming less than 200 MB of GPU memory. On ImageNet, it only requires 6GB of memory and roughly 10 minutes (see Fig. 4). Remarkably, despite such modest compute and memory requirements, R-ESC still achieves SOTA performance and strong robustness. These traits underscore the scalability of our framework to larger datasets and models, making it a practical solution for real-world deployment.

4.3 Concept Erasure on Diffusion Models

Artistic Style Erasure. For the artistic style erasure task, which has emerged as a standard benchmark for testing concept-level erasure in generative models, we follow prior works [10, 27] and construct an evaluation set using 20 prompts for each of 10 artists: 5 classical (Van Gogh, Pablo Picasso, Rembrandt, Andy Warhol, Caravaggio) and 5 modern (Kelly McKernan, Thomas Kinkade, Tyler Edlin, Kilian Eng, and the anime series Ajin: DemiHuman), all of whom are reported to be mimicked by SD. We apply R-ESC to remove two artistic styles: Van Gogh and Kelly McKernan. Erasure performance is measured via LPIPS scores [28], comparing outputs before and after erasure to assess visual similarity. We report three metrics: L_f (score on forget artists, higher is better), L_r (score on remain artists, lower is better), and their overall tradeoff L_d = L_f - L_r.

As shown in Table 3, our proposed method achieves highly competitive performance across all three metrics: demonstrating strong erasure of the forget style (high L_f), minimal distortion to remain styles (low L_r), and the best overall tradeoff (highest L_d). Qualitative results shown in Fig. 5 further support these findings. Notably, ours is the only method that successfully removes Van Gogh's iconic artistic style while faithfully adhering to the input prompt, generating a coherent image of a starry night over a town without reproducing the signature spiral patterns or brush stroke textures.

Table 3: LPIPS scores for artistic style erasure for different unlearning methods.

	Van Gogh			Kelly McKernan		
Method	**L_f (↑)**	**L_r (↓)**	**L_d (↑)**	**L_f (↑)**	**L_r (↓)**	**L_d (↑)**
CA	0.3	0.13	0.17	0.22	0.17	0.05
ESD	0.4	0.26	0.14	0.37	0.21	0.16
SLD-Med	0.31	0.55	-0.24	0.39	0.47	-0.08
SAFREE	0.42	0.31	0.11	0.4	0.39	0.01
RECE	0.31	0.08	0.23	0.29	0.04	0.25
UCE	0.25	0.05	0.2	0.25	0.03	0.22
Ours	0.33	0.08	**0.25**	0.33	0.07	**0.26**

Object Erasure. We follow the evaluation protocol established by UCE [6], assessing the model's ability to erase specific classes from the Imagenette dataset [13] (e.g., "Church", "Cassette Player", "Tench"). We generate images using class-specific prompts and evaluate the generated samples using a pretrained ResNet-50 classifier. We report two primary metrics: the Accuracy of Erased Class (lower is better, indicating effective removal of the concept) and the Accuracy of Other Classes (higher is better, indicating preservation of unrelated concepts).

Table 4: Quantitative results for object erasure on Imagenette using SD v1.4. We report the Accuracy of the Erased Class (lower is better) and the Accuracy of Other Classes (higher is better).

	Acc. of Erased Class (↓)				Acc. of Other Classes (↑)			
Class name	**SD**	**ESD-u**	**UCE**	**Ours**	**SD**	**ESD-u**	**UCE**	**Ours**
Cassette Player	15.6	0.6	0.0	0.0	85.1	64.5	90.3	84.0
Chain Saw	66.0	6.0	0.0	0.0	79.6	68.2	76.1	79.6
Church	73.8	54.2	8.4	0.6	78.7	71.6	80.2	76.6
Gas Pump	75.4	8.6	0.0	2.0	78.5	66.5	80.7	76.5
Tench	78.4	9.6	0.0	0.4	78.2	66.6	79.3	74.2
Garbage Truck	85.4	10.4	14.8	0.0	77.4	51.5	78.7	75.6
English Springer	92.5	6.2	0.2	0.0	76.6	62.6	78.9	78.0
Golf Ball	97.4	5.8	0.8	0.0	76.1	65.6	79.0	75.4
Parachute	98.0	23.8	1.4	0.0	76.0	65.4	77.4	74.7
French Horn	99.6	0.4	0.0	1.0	75.8	49.4	77.0	71.4
Average	78.2	12.6	<u>2.6</u>	**0.4**	78.2	63.2	**79.8**	<u>76.6</u>

Furthermore, Table 4 reports the quantitative evaluation for the object erasure task. Our method achieves near-perfect erasure, reducing the average accuracy of the erased classes uniformly down to exactly 0.4%—a substantial improvement over competitive baselines. While this strong erasure comes with a slight drop in the accuracy of other classes (76.6%) compared to UCE (79.8%), it is important to note that our method still outperforms ESD-u overall and fundamentally eliminates the target concepts with remarkable precision. This highlights the generality of our work beyond classification and artistic styles, effectively translating robust feature-level erasure to complex generative concept removal. We provide additional qualitative results and discussion in Appendix L.

Table 5: Artistic style erasure on SD 2.1 and SDXL. L_f/L_r/L_d: LPIPS on the forget style / remain styles / their difference (higher L_d is better). Best in **bold**, second-best <u>underlined</u>; R-ESC rows shaded.

		Van Gogh			K. McKernan		
	Method	L_f ↑	L_r ↓	L_d ↑	L_f ↑	L_r ↓	L_d ↑
SD2.1	UCE	0.31	<u>0.02</u>	<u>+0.29</u>	0.09	<u>0.04</u>	+0.05
	RECE	**0.33**	0.05	+0.28	**0.12**	0.05	<u>+0.07</u>
	Ours	0.31	**0.01**	**+0.30**	<u>0.11</u>	**0.02**	**+0.09**
SDXL	UCE	0.13	<u>0.01</u>	<u>+0.11</u>	<u>0.15</u>	0.03	+0.11
	RECE	<u>0.15</u>	0.07	+0.09	**0.16**	<u>0.03</u>	**+0.13**
	Ours	**0.16**	**0.01**	**+0.15**	0.12	**0.01**	<u>+0.12</u>

Scaling to Larger Diffusion Models. We repeat the artistic-style and object-erasure evaluations of Tables 3 and 4 on the larger SD 2.1 and SDXL backbones, keeping the protocol identical. Tables 5 and 6 report the results. For artistic-style erasure, R-ESC consistently attains the best overall tradeoff (highest L_d) on both models and both artists, indicating that it removes the target style while leaving remain styles essentially untouched. The gap is starkest for object erasure

Table 6: Object erasure on SD 2.1 and SDXL, averaged over 10 Imagenette classes. A_e/A_r/A_d: ResNet-50 top-1 accuracy on the erased class / retained 9 classes / their difference. Best in **bold**, second-best <u>underlined</u>; R-ESC rows shaded.

	SD 2.1			*SDXL*		
Method	A_e ↓	A_r ↑	A_d ↑	A_e ↓	A_r ↑	A_d ↑
UCE	<u>17.9</u>	<u>48.9</u>	<u>+31.0</u>	**0.0**	0.0	+0.0
RECE	**0.0**	0.3	+0.3	**0.0**	0.0	+0.0
Ours	26.6	**77.6**	**+51.0**	<u>67.8</u>	**86.4**	**+18.6**

(Table 6): the closed-form baselines degrade on SD 2.1 and collapse on SDXL (UCE and RECE drive both accuracies to zero), whereas R-ESC retains high accuracy and a large positive gap (A_d) on both. R-ESC thus scales to larger diffusion models without architecture-specific tuning.

4.4 Ablation Study

Effectiveness of PO Projection. We perform ablation studies on the key components of our framework: PO projection, erasing, remapping, and the full R-ESC architecture. Table 2 summarizes results on CIFAR-10, while additional results on CIFAR-100 are provided in the Appendix §E.2. Here, **Erase** refers to an ESC-like baseline that simply removes forget prototypes, while **Remap** denotes our proposed remapping strategy. **PO** indicates whether PO projection is applied. Without PO, erase fails to fully forget, with forget accuracy lingering around 14%. Remap can drive forget accuracy to zero, but at the cost of remain accuracy degradation.

Applying PO prior to erasure yields the strongest results. By decorrelating prototype vectors, PO ensures that erasure and remapping precisely target forget prototypes without corrupting remain prototypes. Empirical evidence for this effect is shown in Fig. 6, which plots cosine similarities between prototypes after erasing and remapping. In contrast to Fig. 3, where erasure perturbed remain prototypes, here remain prototypes preserve their autocorrelation close to one (diagonal entries). Meanwhile, the forget prototype exhibits a cosine similarity of 0 with itself under erasure, or a similarity close to 1 with the remain prototype of class 0 under remapping. These results validate the role of PO in disentangling prototypes and enabling precise editing of forget prototypes.

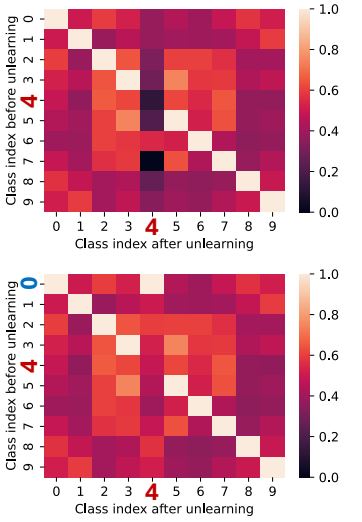

Fig. 6: Cosine similarities between the mean activations after erasing (*top*) and remapping (*bottom*). Class 4 is unlearned and remapped to class 0.

Sensitivity Analysis for Target Remapping Class. We also study the effect of the target prototype used for remapping (see Table 7). Using CIFAR-10 with class 4 as the forget class, we remap it to different remain classes. Performance remains strong across targets, though some yield slightly better results, suggesting mild preference. We leave deeper investigation to future work.

Sensitivity Analysis for Number of Experts. Fig. 7 (detailed in Appendix §E.3) shows sensitivity analysis results with respect to the number of experts evaluated on CIFAR-10, CIFAR-100, and Tiny-ImageNet under the KR setting. HM performance is stable across a wide range of expert counts, indicating its

robustness. The only exception is CIFAR-10 with a single expert, which reduces to basic remapping and causes a dip in performance. More detailed results are provided in the Appendix §E.3.

Stochastic vs. Conditional router. While R-ESC adopts stochastic routing as default for its compute efficiency, we also explored conditional variants, showcasing the potential of optimized routers (see Table 8). **R-ESC (P)** uses pseudo-inverse initialization of remain prototypes, while **R-ESC (P-T-B)** is a trained version with forget load balancing to prevent expert specialization collapse. We can see that the trained routers yield higher HM on CIFAR, but not consistently. Detailed analysis of methods and experiments are provided in the Appendix §G.

Table 7: Unlearning performance measured with different target remapping classes using AllCNN on CIFAR-10.

CIFAR-10						
Target	D_f	D_r	D_ft	D_rt	HM	HM_t
0	0.00	99.87	0.00	91.16	99.94	95.38
1	0.00	99.87	0.00	90.76	99.93	95.16
2	0.00	99.84	0.00	91.06	99.92	95.32
3	0.00	99.75	0.00	90.68	99.87	95.11
5	0.00	99.74	0.00	90.68	99.87	95.11
6	0.00	99.77	0.00	90.94	99.88	95.26
7	0.00	99.85	0.00	91.11	99.93	95.35
8	0.00	99.86	0.00	90.76	99.93	95.16
9	0.00	99.84	0.00	90.69	99.92	95.12

CIFAR-10 (KR setting; lr = 0.1)						
Target	D_f	D_r	D_ft	D_rt	HM	HM_t
0	33.20	96.17	29.83	87.76	66.89	69.78
1	33.27	96.20	30.03	87.36	66.75	69.34
2	66.29	92.48	62.00	84.46	34.06	40.38
3	36.40	95.63	34.23	87.34	65.00	66.02
5	33.51	96.14	31.27	87.69	66.67	67.39
6	33.26	96.14	31.03	87.49	66.75	67.77
7	66.49	92.50	61.33	84.19	33.67	41.49
8	67.05	92.49	60.60	83.93	33.42	43.37
9	89.95	89.62	80.90	81.34	15.24	29.26

Finally, to assess generalizability, we applied CE at shallower layers (Table 9). It remains effective at the second-last layer but weakens at the third-last, highlighting sensitivity to layer choice and a promising avenue for future study. For full experimental details and additional results not highlighted in the main text—including comprehensive KR results—we refer readers to the Appendix §E.4.

Table 8: Ablation table for stochastic random routing (R-ESC) and conditional routing

CIFAR-10 (N=8, KR setting; lr = 0.1)						
Method	D_f	D_r	D_ft	D_rt	HM	HM_t
R-ESC	5.25 ± 3.77	96.35 ± 3.09	13.73 ± 6.64	84.27 ± 3.48	95.54 ± 3.42	85.24 ± 5.04
R-ESC (P)	5.89 ± 5.34	96.94 ± 2.04	6.33 ± 5.34	87.31 ± 2.05	95.47 ± 4.37	90.35 ± 4.01
R-ESC (P-T-B)	0.97 ± 0.37	95.48 ± 0.84	1.7 ± 0.61	86.08 ± 0.85	**97.22 ± 0.61**	**91.79 ± 0.74**
CIFAR-100 (N=80, KR setting; lr = 0.1)						
Method	D_f	D_r	D_ft	D_rt	HM	HM_t
R-ESC	0.03 ± 0.02	99.98 ± 0.00	1.90 ± 0.87	71.00 ± 0.19	**99.97 ± 0.01**	82.37 ± 0.30
R-ESC (P)	9.53 ± 1.09	98.13 ± 0.44	10.70 ± 0.75	68.15 ± 0.65	94.14 ± 0.71	81.25 ± 0.63
R-ESC (P-T-B)	4.53 ± 1.46	99.31 ± 0.16	6.30 ± 0.82	73.67 ± 0.42	97.34 ± 0.69	**82.48 ± 0.56**

4.5 Other Experiments

Due to space constraints, we defer several additional experiments and broader analyses to the Appendix. These include two further robustness studies that probe irreversibility from complementary angles: an adversarial-prompt red-teaming evaluation of R-ESC on diffusion models under the UnlearnDiff, P4D, and Ring-A-Bell attacks (§J), and a retain-only fine-tuning attack on the unlearned classifier (§K). We also evaluate R-ESC

Table 9: Unlearning performance when the proposed methods are applied to shallower layers.

	CIFAR-10						
	Method	D_f	D_r	D_ft	D_rt	HM	HM_t
2nd last	Erase	0.95 ± 0.02	99.90 ± 0.00	0.50 ± 0.00	91.37 ± 0.01	99.47 ± 0.01	95.26 ± 0.00
	Remap	0.44 ± 0.00	99.52 ± 0.00	0.13 ± 0.06	90.53 ± 0.01	99.54 ± 0.00	94.97 ± 0.03
	R-ESC	0.67 ± 0.02	99.60 ± 0.01	0.37 ± 0.06	90.73 ± 0.16	99.47 ± 0.02	94.98 ± 0.11
3rd last	Erase	46.30 ± 0.15	98.07 ± 0.01	36.63 ± 0.38	88.87 ± 0.02	69.40 ± 0.13	73.98 ± 0.26
	Remap	24.26 ± 0.16	96.22 ± 0.05	18.43 ± 0.12	86.90 ± 0.04	84.76 ± 0.11	84.15 ± 0.08
	R-ESC	25.28 ± 0.24	96.36 ± 0.09	19.47 ± 0.45	87.31 ± 0.10	84.17 ± 0.18	83.78 ± 0.25
	CIFAR-100						
	Method	D_f	D_r	D_ft	D_rt	HM	HM_t
2nd last	Erase	5.20 ± 0.20	99.98 ± 0.00	0.60 ± 0.00	81.49 ± 0.01	97.32 ± 0.11	89.56 ± 0.00
	Remap	0.13 ± 0.12	99.98 ± 0.00	0.00 ± 0.00	80.71 ± 0.00	99.92 ± 0.06	89.33 ± 0.00
	R-ESC	0.20 ± 0.20	99.98 ± 0.00	0.00 ± 0.00	80.38 ± 0.03	99.89 ± 0.10	89.12 ± 0.02
3rd last	Erase	12.53 ± 1.33	99.78 ± 0.00	1.90 ± 0.00	79.13 ± 0.00	93.22 ± 0.75	87.60 ± 0.00
	Remap	8.67 ± 2.01	99.04 ± 0.00	1.60 ± 0.00	77.62 ± 0.01	95.02 ± 1.09	86.78 ± 0.00
	R-ESC	10.00 ± 2.36	99.20 ± 0.01	2.07 ± 0.21	77.91 ± 0.17	94.36 ± 1.31	86.78 ± 0.18

on a random data forgetting task (§F). We also provide complete utility and Knowledge Retention (KR) evaluation results, where we extend our main KR evaluations across a wide range of learning rates (0.001, 0.01, and 0.1) (§E.1). Additionally, we supply comprehensive ablation studies that expand our analysis of the PO projection to CIFAR-100 and Tiny-ImageNet (§E.2), and provide detailed metrics to assess the method's sensitivity to varying the number of remapping experts (§E.3). We encourage readers to refer to the Appendix for these supplementary experiments, which offer deeper insights into the performance and scope of our approach.

5 Conclusion

We introduced R-ESC (Robustly Erasing Space Concepts), a training-free framework for robust feature-level CE that combines prototype-orthogonal projection and expert-based remapping. Across diverse datasets and architectures, R-ESC consistently outperforms existing methods, surpassing training-based baselines at a fraction of the cost. Unlike prior approaches that allow recovery of forget knowledge through linear probing, R-ESC maintains accuracy near random-guess levels, delivering robust guarantees stronger than the traditional retrain-from-scratch benchmark. These results position R-ESC as both effective and efficient, paving the way for scalable deployment in safety-critical systems. Beyond immediate effectiveness, R-ESC opens entirely new avenues for research: how to optimize routers for erasure experts, the potential for hybrid conditional-stochastic routing, and refined prototype identification strategies. By enabling both SOTA performance and rich opportunities for exploration, R-ESC establishes a new standard for CE research.

Acknowledgements

This work was supported in part by the National Research Foundation of Korea (NRF) grants funded by the Korea government (MSIT) (No. RS-2024-00345732; No. RS-2025-02216217); the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No. RS-2025-25442569, AI Star Fellowship Support Program (Sungkyunkwan Univ.); No. IITP-2026-RS-2020-II201821, ICT Creative Consilience Program; No. RS-2021-II212052, Information Technology Research Center (ITRC); No. RS-2019-II190421, AI Graduate School Support Program (Sungkyunkwan University); No. RS-2019-II190079, Artificial Intelligence Graduate School Program (Korea University); and No. RS-2024-00457882, AI Research Hub Project); the "Advanced GPU Utilization Support Program" funded by the Government of the Republic of Korea (Ministry of Science and ICT); and the InnoCORE program of the Ministry of Science and ICT (No. N10250156).

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C.A., Jia, H., Travers, A., Zhang, B., Lie, D., Papernot, N.: Machine unlearning. In: 2021 IEEE Symposium on Security and Privacy (SP). pp. 141–159. IEEE (2021)
3. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021)
4. Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., et al.: Toy models of superposition. arXiv preprint arXiv:2209.10652 (2022)
5. Gandikota, R., Materzynska, J., Fiotto-Kaufman, J., Bau, D.: Erasing concepts from diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2426–2436 (2023)
6. Gandikota, R., Orgad, H., Belinkov, Y., Materzyńska, J., Bau, D.: Unified concept editing in diffusion models. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 5111–5120 (2024)
7. Gao, H., Pang, T., Du, C., Hu, T., Deng, Z., Lin, M.: Meta-unlearning on diffusion models: Preventing relearning unlearned concepts (2025)
8. Ginart, A., Guan, M., Valiant, G., Zou, J.Y.: Making ai forget you: Data deletion in machine learning. In: Advances in Neural Information Processing Systems. vol. 32 (2019)
9. Goel, S., Prabhu, A., Kumaraguru, P.: Evaluating inexact unlearning requires revisiting forgetting. CoRR (2022)
10. Gong, C., Chen, K., Wei, Z., Chen, J., Jiang, Y.G.: Reliable and efficient concept erasure of text-to-image diffusion models. In: European Conference on Computer Vision. pp. 73–88. Springer (2024)
11. Graves, L., Nagisetty, V., Ganesh, V.: Amnesiac machine learning. In: AAAI Conference on Artificial Intelligence (2021)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016)
13. Howard, J.: Imagenette: A smaller subset of 10 easily classified classes from imagenet (2019)
14. Kim, Y., Cha, S., Kim, D.: Are we truly forgetting? a critical re-examination of machine unlearning evaluation protocols. arXiv preprint arXiv:2503.06991 (2025)
15. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech. rep., University of Toronto (2009)
16. Kumari, N., Zhang, B., Wang, S.Y., Shechtman, E., Zhang, R., Zhu, J.Y.: Ablating concepts in text-to-image diffusion models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22691–22702 (2023)
17. Kurmanji, M., Triantafillou, P., Hayes, J., Triantafillou, E.: Towards unbounded machine unlearning. Advances in neural information processing systems pp. 1957–1987 (2023)
18. Le, Y., Yang, X.: Tiny imagenet visual recognition challenge. CS231N Course Report (2015)

19. Lee, T.Y., Park, S., Jeon, M., Hwang, H., Park, G.M.: Esc: Erasing space concept for knowledge deletion. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5010–5019 (2025)
20. Nguyen, T.T., Huynh, T.T., Ren, Z., Nguyen, P.L., Liew, A.W.C., Yin, H., Nguyen, Q.V.H.: A survey of machine unlearning. arXiv preprint arXiv:2209.02299 (2024)
21. Regulation, P.: Eu general data protection regulation (2016)
22. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
23. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision **115**(3), 211–252 (2015)
24. Schramowski, P., Brack, M., Deiseroth, B., Kersting, K.: Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22522–22531 (2023)
25. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.: Striving for simplicity: The all convolutional net. In: ICLR Workshop (2015)
26. Thudi, A., Jia, H., Shumailov, I., Papernot, N.: On the necessity of auditable algorithmic definitions for machine unlearning. In: 31st USENIX security symposium (USENIX Security 22). pp. 4007–4022 (2022)
27. Yoon, J., Yu, S., Patil, V., Yao, H., Bansal, M.: Safree: Training-free and adaptive guard for safe text-to-image and video generation. In: ICLR 2025 (2025)
28. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)