This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Reminiscence Attack on Residuals: Exploiting Approximate Machine Unlearning for Privacy

Yaxin Xiao[§] 20034165r@connect.polyu.hk
Qingqing Ye[§] qqing.ye@polyu.edu.hk
Li Hu[§] lily23.hu@polyu.edu.hk
Huadi Zheng[†] zhenghuadi@huawei.com
Haibo Hu[§‡*] haibo.hu@polyu.edu.hk
Zi Liang[§] zi1415926.liang@connect.polyu.hk
Haoyang Li[§] hao-yang9905.li@connect.polyu.hk
Yijie Jiao[§] 22103037d@connect.polyu.hk

[§]Department of Electronical and Electronic Engineering, The Hong Kong Polytechnic University [†]Huawei Technologies Co., Ltd.
[‡]Research Centre for Privacy and Security Technologies in Future Smart Systems [*]Corresponding author: haibo.hu@polyu.edu.hk

Abstract

Machine unlearning enables the removal of specific data from ML models to uphold the ***right to be forgotten****. While approximate unlearning algorithms offer efficient alternatives to full retraining, this work reveals that they fail to adequately protect the privacy of unlearned data. In particular, these algorithms introduce implicit residuals which facilitate privacy attacks targeting at unlearned data. We observe that these residuals persist regardless of model architectures, parameters, and unlearning algorithms, exposing a new attack surface beyond conventional output-based leakage. Based on this insight, we propose the* ***Reminiscence Attack (ReA)****, which amplifies the correlation between residuals and membership privacy through targeted fine-tuning processes. ReA achieves up to* $1.90\times$ *and* $1.12\times$ *higher accuracy than prior attacks when inferring class-wise and sample-wise membership, respectively. To mitigate such residual-induced privacy risk, we develop a dual-phase approximate unlearning framework that first eliminates deep-layer unlearned data traces and then enforces convergence stability to prevent models from "pseudo-convergence", where their outputs are similar to retrained models but still preserve unlearned residuals. Our framework works for both classification and generation tasks. Experimental evaluations confirm that our approach maintains high unlearning efficacy, while reducing the adaptive privacy attack accuracy to nearly random guess, at the computational cost of* $2-12\%$ *of full retraining from scratch.*

1. Introduction

As the significance of individual privacy grows, numerous laws and regulations have been established to grant individuals the right to protect their privacy. Notably, the General Data Protection Regulation (GDPR) [33] and the Personal Information Protection and Electronic Documents Act (PIPEDA) [1] incorporate the *right to be forgotten*, requiring organizations to delete user data upon request. In addition to traditional database systems, machine learning (ML) systems, which are trained on vast amounts of private data, must comply with this right by removing data from their training sets. This is critical because ML systems are vulnerable to data privacy breaches, such as membership inference attacks [39, 46] and model inversion attacks [29].

The techniques that support such data removal in ML systems are collectively called machine unlearning (MU) [6]. State-of-the-art (SOTA) machine unlearning methods can be categorized into exact unlearning and approximate unlearning. Exact unlearning necessitates retraining the model from scratch using only the remaining dataset, which is resource-intensive even with optimization techniques [6]. In contrast, approximate machine unlearning (AMU) modifies the parameters of pre-trained machine learning models to approximate the retained model, allowing selective forgetting while maintaining efficiency. Currently, various AMU methods [10, 17, 24, 25, 45, 48] have achieved near-retrained unlearning efficacy when measured by legacy metrics such as prediction accuracy [25]. Due to their significantly lower resource demands, AMU algorithms have become a focal point in this field.

However, while machine unlearning initially aims to protect the privacy of the target data, it may inadvertently introduce new privacy risks. Such risks come from the imprint of unlearned data left behind in unlearned models, which becomes a source of privacy leakage. According to the imprint types, existing privacy attacks against unlearned data can be divided into two categories: version-dependent and

version-independent attacks. Version-dependent privacy attacks exploit updates between historical model versions to infer the privacy of unlearned data [5, 9, 22]. Thus, they are applicable only in scenarios where historical model versions are released. The second type, version-independent attacks, applies to broader scenarios and hence serves as the focus of this study. These attacks [28] detect privacy leaks by analyzing differences in the model's behavior on unlearned data versus non-training data. Yet, we argue that current version-independent methods [7, 28] only partially exploit the problem.

This paper investigates a critical oversight in version-independent privacy leakage analysis: Do existing approximate machine unlearning (AMU) methods inadvertently leak unlearned data privacy through **implicit residuals beyond the unlearned model's output distributions**? Prior works [7, 28] focus on output-space residuals yet fail to capture the full extent of privacy risks. We reveal that AMU methods overlook latent residuals that alter their loss landscapes, leading to even greater risks for the membership privacy of unlearned data. To exploit these residuals, we introduce the Reminiscence Attack (ReA), which leverages a targeted fine-tuning approach to detect whether specific classes or samples were part of the training set.

To address such privacy leakages, we propose Orthogonal Unlearning & Replay (OUR), an AMU method designed to scrub latent residuals in the model. OUR is built on two key insights: First, class-wise residuals persist because the unlearned model retains high intra-class correlation in hidden representations for the unlearned class, even when its predictions resemble those of retrained models. For deeper scrubbing, we enforce orthogonality between the hidden representations of unlearned data and their original values. Second, sample-wise residuals arise because the unlearned model stays in a "pseudo-converged" states that mimic the retrained model outputs (e.g., the prediction accuracy) but preserves latent information of unlearned data. To counteract this, we separate OUR into two phases to ensure model convergence through a final fine-tuning phase. In summary, our contributions are as follows:

- We introduce *Reminiscence Attack* (ReA), a novel version-independent attack which targets the membership privacy of unlearned samples and classes. ReA uncovers privacy leakage beyond what existing attacks [25, 28] detect by exploiting residuals in the model's loss landscape.
- To address the privacy vulnerabilities exposed by ReA, we propose an approximate unlearning algorithm, Over-Unlearning & Replay (OUR), designed to eliminate latent residuals within the unlearned models.
- Extensive experiments evaluate the unlearning efficacy of approximate unlearning algorithms and their privacy leakage using our ReA attacks, with results in classification scenarios summarized in Figure 1. ReA effectively exposes unlearned data privacy, achieving up to 90% attack accuracy for unlearned class membership inference. The results also confirm that OUR significantly reduces ReA's attack accuracy to near-random levels while preserving high unlearning efficacy and efficiency.

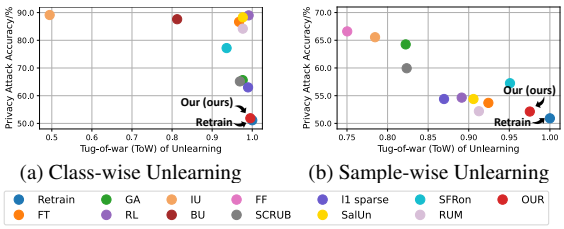


Figure 1. Comparison of unlearning efficacy (Tug-of-War metric, Equation 1) and ReA attack accuracy across 12 unlearning methods tested in Section 5.2 on CIFAR20 (11 baselines and our OUR). OUR method approaches the "Retrain" in the down-right corner, showing high unlearning efficacy and low privacy leakage.

2. Problem Definition

2.1. Machine Unlearning (MU)

Machine unlearning (MU) removes the influence of specific training data on a pre-trained machine learning model $F(\cdot;\theta)$ through exact or approximate methods. While exact unlearning requires retraining from scratch, approximate unlearning modifies model parameters θ to achieve comparable unlearning efficacy at significantly lower computational cost. Formally, given a model $F(\cdot;\theta)$ trained on dataset $\mathcal{D}$, let $\mathcal{D}_u$ denote unlearned data and $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_u$ to the retained data. The approximate unlearning algorithm $\mathcal{U}$ aims to produce parameters $\theta^u = \mathcal{U}(\theta)$ such that $F(\cdot;\theta^u)$ approximates the model $F(\cdot;\theta^0)$ retrained on $\mathcal{D}_r$.

Unlearning in Classification. In classification tasks, unlearning operates at two granularities: **sample-wise unlearning** for individual data points and **class-wise unlearning** for entire classes [17]. Specifically, sample-wise unlearning processes $\mathcal{D}_u$ as discrete samples, requiring the unlearned model $F(\cdot;\theta^u)$ to match the retrained model $F(\cdot;\theta^0)$'s behavior by treating $\mathcal{D}_u$ as unseen test data. Class-wise unlearning handles $\mathcal{D}_u$ as entire classes, and demands $F(\cdot;\theta^u)$ aligns with $F(\cdot;\theta^0)$ to recognize unlearned classes as out-of-distribution (OOD) classes. This ensures the unlearned model has identical task scopes to $F(\cdot;\theta^0)$.

Unlearning in Image Generation. For generative models such as text-to-image diffusion models [36], we shift focus from safety alignment to privacy preservation. While existing methods focus on removing harmful contents (e.g., nudity) [13, 15], we address **identity-specific concept unlearning** in portrait generation. This process aims to eliminate identity-specific concepts from $F(\cdot,\theta)$ while maintaining output quality for retained tasks.

Legacy MU Metrics. Machine unlearning is assessed with

the following legacy metrics [25, 30, 45]: ❶ **Test Accuracy (TA)**. It is defined as TA= $Acc_{\mathcal{D}_t}(F(.;\theta^u))$, which measures the accuracy of $F(.;\theta^u)$ on the test dataset $\mathcal{D}_t$. ❷ **Unlearning Accuracy (UA)**. It is calculated by UA= $Acc_{\mathcal{D}_u}(F(.;\theta^u))$. ❸ **Retaining Accuracy (RA)**. It's defined as RA= $Acc_{\mathcal{D}_r}(F(.;\theta^u))$. The closer the three accuracy metrics are to the 'ground-truth' retrained model $F(.;\theta^0)$, the better the unlearning efficacy. ❹ **Membership Inference Attack on $\mathcal{D}_u$ (MIA Efficacy)**. This metric quantifies the success rate of membership inference attacks [39] in detecting $\mathcal{D}_u$ as training data for $F(\cdot;\theta^u)$. Unlike prior evaluations [25] measuring only the true positive rate (TPR), we adopt TPR at a fixed 0.1 false positive rate (FPR) for stricter privacy assessment. ❺ **Run-Time Efficacy (RTE)**. This metric refers to the time a machine unlearning algorithm consumes, which assesses its practical efficiency. ❻ **Tug-of-War (ToW)** [48]. This unified metric quantifies the alignment between unlearned (θ^u) and fully retrained (θ^0) models across multiple metrics M_1, M_2, and M_3. When instantiated with TA, UA, and RA, ToW measures unlearning efficacy. Let $d(\cdot,\cdot)$ denote the absolute distance between metric values. The ToW score is defined as:

$$\text{ToW}(M_1, M_2, M_3) = \prod_{i=1}^{3}(1 - \frac{d(M_i(\theta^u), M_i(\theta^0))}{M_i(\theta^0)}). \quad (1)$$

2.2. Privacy Attacks Targeting MU

Privacy attacks compromise unlearned data privacy via exploring information leakage in machine unlearning processes. These attacks fall into two categories: (1) version-dependent attacks, which analyze model update histories [5, 9, 22], and (2) version-independent attacks, which leverage residuals of approximate unlearning. As version management addresses only the former, we focus on the latter category which exposes vulnerabilities in approximate unlearning methods. This category includes two benchmarks: (1) **MIA-LiRA**. It applies SOTA membership inference attacks (MIA) LiRA [3, 7], which uses shadow models and a likelihood ratio test on model confidence to detect unlearned data. Since its attack model is designed to detect training data for the original model $F(\cdot;\theta)$, its effectiveness diminishes when unlearned and training data yield distinct output distributions. (2) **MIA against User Privacy (MIA-UP)** [28]. This attack distinguishes unlearned data from test data. While adopting LiRA's shadow model framework, this method uniquely synchronizes shadow models with the unlearned model's unlearning process. Then, the binary classifier attacker trains on the outputs of shadow models, using unlearned data as positives and test data as negatives. However, it becomes unstable if shadow and unlearned models' unlearning processes diverge. Up to now, privacy attacks targeting unlearned data primarily exploit residuals in output distributions.

2.3. Threat Model

While existing privacy attacks on machine unlearning focus on sample-level membership inference, we pioneer the formalization of **class membership privacy** leaks, which infer whether specific classes were ever unlearned. We assume adversaries are aware of the task scope and have either white-box or black-box API access.

Sample-wise Membership Inference Attack (MIA). For models undergoing sample-wise unlearning ($F(;\theta^u)$), adversaries infer membership of individual unlearned samples, where test data is negative references. Following SOTA MIA evaluations [4, 7], we analyze the true-positive rate (TPR) versus the false-positive rate (FPR) of attacks (i.e., the precision-recall tradeoff) rather than fixed accuracy. Specifically, the attack model outputs continous confidence scores through $\mathcal{A}'$, which are then thresholded by τ to yeild a membership prediction. Formally, for an inferred sample x with label y, the output of the sample-MIA attacker $\mathcal{A}_{\text{sample}}$ is:

$$\mathcal{A}_{\text{sample}}(x, y) = \mathbf{1}[\mathcal{A}'(F(x;\theta^u), y) > \tau], \quad (2)$$

where $\mathbf{1}$ is the indicator function and τ controls the precision-recall tradeoff.

Class-wise Membership Inference Attack (MIA). For models undergoing $F(.;\theta^u)$ undergoes class-wise unlearning, the adversary infers membership of candidate OOD classes $\mathcal{D}^{\text{ood}}$, i.e., determining whether it was unlearned. Crucially, this attack can extend to concept unlearning in generative tasks, targeting the membership of forgotten concepts. Mirroring the sample-wise formulation (Equation 2), the output of the class-wise MIA attacker $\mathcal{A}_{\text{class}}$ for $\mathcal{D}^{\text{ood}}$ is:

$$\mathcal{A}_{\text{class}}(\mathcal{D}^{\text{ood}}) = \mathbf{1}[\mathcal{A}'(\mathcal{D}^{\text{ood}}) > \tau]. \quad (3)$$

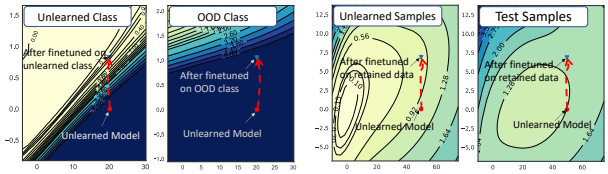
3. Reminisence Attack (ReA): Exploiting Loss Landscape Residuals

We propose the Reminiscence Attack (ReA), a novel version-independent membership inference attack against unlearned samples, classes, and concepts. This method exploits latent residuals preserved in the loss landscape during approximate unlearning, establishing an orthogonal attack surface to existing approaches (MIA-LiRA/UP).

Understanding Unlearning Residuals in Loss Landscapes. Approximate unlearning may produce measurable residuals that deviate from 'gold-standard' retrained models, causing persistent privacy risks. Formally, for an unlearned model $F(\cdot;\theta^u)$ obtained by unlearning dataset $\mathcal{D}_u$, we define its residual as

$$R(\mathcal{D}_u) := \mathbb{E}\left[d(F(\mathcal{D}_u;\theta^u), F(\mathcal{D}_{\text{non-tr}};\theta^u))\right], \quad (4)$$

where $\mathcal{D}_{\text{non-tr}}$ is the non-training counterpart to $\mathcal{D}_u$. d is a task-specific metric measuring model behavior discrepancy. We reveal that these residuals not only manifest in output distributions but also reshape the loss landscape. As visualized in Figure 2a, unlearned models remain near optima



(a) Class-wise Unlearn (RUM [48]) (b) Sample-wise Unlearn (RL [18])

Figure 2. Visualized loss landscapes around unlearned models in CIFAR20 experiments. Dashed lines are guided fine-tuning trajectories. Appendix C provides additional results on other datasets.

(bright regions) for $\mathcal{D}_u$ data and converge rapidly under guided finetuning, revealing latent traces, contrasting with non-training data (including OOD classes and test samples).

To efficiently expose membership information in the loss landscape, targeted fine-tuning is the natural choice, which we term **reminiscence**. Its implementation depends on the inference target. In class-wise unlearning, as shown in Figure 2a (left), the unlearned model lies on flat plateaus near sharp minima. Guiding fine-tuning toward the minima accelerates convergence for unlearned classes while slowing non-training classes, which reveals membership. This guidance is achieved by introducing retained data in fine-tuning, as the targeted path aligns with their optimal solutions.

In sample-wise unlearning, as shown in Figure 2b (left), the unlearned model remains on a steep loss landscape for unlearned data, indicating incomplete convergence. If the model shifts to the low-loss basin (bright regions), the loss gap between unlearned and test data widens, further exposing membership. Unlike class-wise ReA, inferred samples do not share memberships and cannot be fine-tuned in groups. However, fine-tuning on retained samples still guides the model to the low-loss basin, as their optimal solution regions align with unlearned samples. While both scenarios rely on retained samples, Section 3.1 explains how to remove this assumption.

3.1. Reminiscence Attack (ReA)

Earlier, we introduced the method of reminiscence (targeted fine-tuning) to leverage residuals in the loss landscape to amplify the membership leakage of unlearned data. Here, we detail the steps of the reminiscence attack (ReA).

Class-wise ReA. To detect the link between residuals and an unlearned class, ReA reintroduces the inferred class (a set $\mathcal{D}^{\text{ood}}$) into training and measures how quickly it resonates with residuals to derive the attack confidence score. Formally, this is quantified by the resonance index Idxr, defined as the number of iterations required for the model to achieve high prediction accuracy (i.e., convergence threshold) on $\mathcal{D}^{\text{ood}}$. Then, the confidence score in Equation 3 is computed as:

$$\mathcal{A}'(\mathcal{D}^{\text{ood}}) = 1 - \text{Idx}_{\text{r}}(\mathcal{D}^{\text{ood}})/\text{Idx}_{\text{max}}, \tag{5}$$

where Idx_{max} is the total number of training iterations. A lower Idxr means faster convergence and suggests the inferred class closer to the unlearned class in Figure 2a, which leads to a higher confidence score in Equation 5 when inferred as unlearned data. The above method assumes a white-box attacker, but this assumption can be relaxed with substitute models from model extraction [41], given the demonstrated transferability of residuals in Appendix E.8.

There are two key challenges affecting class-wise ReA's efficiency. First, ReA ideally requires the retained dataset to guide the shortest optimal path, but attackers may lack access to it. This issue can be mitigated by minimizing logits updates from OOD samples (excluding the inferred class) [31], as this fulfills the retained dataset's key role in maintaining utility. Second, the fine-tuned learning rate (lr) significantly impacts ReA's stability. To address this, we introduce the **multi-lr aggregation strategy**, which averages resonance indexes across multiple learning rates $\{\text{lr}_j\}_{j=1}^{J}$. Thus, the revised confidence score is:

$$\mathcal{A}'(\mathcal{D}^{\text{ood}}) = 1 - \left[\sum\nolimits_{j=0}^{J} \text{Idx}_{\text{r}}(\mathcal{D}^{\text{ood}}, \text{lr}_j)/(J \cdot \text{Idx}_{\text{max}})\right]. \tag{6}$$

Sample-wise ReA. In sample-wise unlearning, residuals persist primarily due to incomplete convergence. To fully expose the membership of unlearned samples, ReA fine-tunes the unlearned model on the retained dataset with a reduced learning rate for convergence. If residuals remain, the model's accuracy on unlearned samples rises abnormally higher than on test data, signaling additional privacy leakage. Building on this insight, ReA follows two steps: (1) **the reminiscence process**, where the unlearned model is fine-tuned on the retained dataset, and (2) **the MIA process**, where membership is inferred using MIA-LiRA [7]. MIA-LiRA is preferred over MIA-UP to avoid dependence on unlearning algorithm knowledge. However, the first step requires knowledge of the retained data identity, which is often inaccessible to attackers. To address this, we propose inferring a set of high-confidence positive samples using MIA-LiRA, and them as a "pseudo" retained dataset in the first step. If white-box access is unavailable, sample-wise ReA reduces to MIA-LiRA. Appendix A provides pseudo codes for class-wise and sample-wise ReA.

4. Scrubbing Residuals in Unlearning: Orthogonal Unlearning & Replay (OUR)

Previous analysis indicates that if residuals remain from approximate machine unlearning (AMU) processes, particularly in the overlooked loss landscape, they may unintentionally facilitate membership inference attacks (e.g., ReA) against unlearned samples. This highlights the necessity to develop AMU methods that not only align output distributions with retrained models but also scrub residuals, to ensure privacy protection for unlearned data.

4.1. High-level Solution of Scrubbing Residuals

To scrub residuals, we must understand their cause and address both class-wise and sample-wise cases. Class-wise

residuals persist when the collective features of the unlearned class remain in deeper hidden layers (far from the output layer), as confirmed in Section 5.3. To eliminate these traces, we enforce orthogonality between the hidden and original features of unlearned samples, achieving deeper scrubbing beyond existing methods (e.g., random label [18]). This process is termed as orthogonal unlearning.

For sample-wise cases, residuals arise from incomplete convergence on retain tasks, where unlearned accuracy (UA) falsely resembles test samples' behavior. Resolving this requires the model to reach an actual convergence state. A single loss function optimizing both and model utility often hinders this process, so we structure unlearning in **two phases**: (1) Orthogonal Unlearning to deeply scrub unlearned sample information, and (2) replay on the retained set to restore convergence and preserve utility. Our notion of orthogonality differs from prior orthogonal projection unlearning [16, 21, 44], which perturbs directions orthogonal to retaining tasks to preserve utility. Instead, we minimize the cosine similarity between unlearned features and their original state to scrub unlearning residuals.

4.2. Orthogonal Unlearning & Replay (OUR)

Next, we detail the two phases of OUR. Let $F_l(:;\theta)$ represent the output of the l-th layer of the unlearned model (i.e., representations/feature), and θ^0 be the pre-unlearning states. In the first phase, the model is trained over e_1 epochs with the following loss function:

$$L_{\text{orth}} = \sum_{(x,y)\in\mathcal{D}_u} \sum_{l\in\{l\}_k} [F_l^T(x;\theta) F_l(x;\theta^0)]^2. \quad (7)$$

L_{orth} encourages orthogonality between the the features of $F(\cdot;\theta)$ and $F(\cdot;\theta^0)$ on unlearned samples. Here, $\{l\}_k$ denotes the set of k hidden layers selected for this phase.

During the replay phase, the unlearned model is fine-tuned on D_r for e_2 epochs, yielding the final unlearned model $F(.;\theta^u)$. To enhance unlearning effectiveness, we incorporate $l1$ regularization optimization [25]. The pseudo-code for OUR is provided in Appendix A.

Key Considerations of OUR. How is run-time efficacy (RTE) maintained? RTE is ensured by excluding the retained set in the orthogonal unlearning phase. While this may induce immediate utility degradation, the replay phase achieves rapid recovery due to the minimal change of neurons. Our analysis in Appendix B shows that most neurons undergo trivial changes during unlearning, allowing efficient recovery in the replay phase.

How to prevent model collapse during orthogonal unlearning? Model collapse occurs when parameter gradients explode. Therefore, the orthogonal unlearning phase should constrain the maximal parameter change $\Delta_{\text{max}} \leq \Delta_{\text{thr}}$. Δ_{thr} is a value (e.g., $5e-3$) which may trigger gradient exposure during replay and set as the maximum distance between two random models (see Appendix B).

5. Experimental Evaluation

In this section, we first evaluate the privacy vulnerabilities of 11 approximate unlearning benchmarks using *Reminiscence Attack* (ReA). We then analyze the unlearning efficacy and privacy leakage of the proposed *Orthogonal Unlearning & Replay* (OUR) framework. Our code is available at `https://github.com/orthogonalunlearning-replay/OUR`.

5.1. Setup

Datasets and Models. We conduct classification experiments on three benchmarks: CIFAR10 with ResNet18, CIFAR100 with ResNet18 [20], and CIFAR20 (20 broad classes aggregated from CIFAR100's fine-grained labels) with Vision Transformer (ViT) [12]. Class-wise unlearning is performed on CIFAR100 and CIFAR20, while sample-wise unlearning applies to CIFAR10 and CIFAR20. In the image-generation domain, we specialize in unlearning for text-to-image Stable Diffusion (SD) [36] with a focus on privacy. Specifically, we perform identity-level unlearning on an SD model fine-tuned via DreamBooth [37] using CelebA-HQ [27] identities.

Unlearning Configuration. We benchmark 6 SOTA AMU algorithms and 5 optimized frameworks (Appendix D.2). For OUR, $\{l\}_k$ uses the first, middle, and last block outputs, as verified in Appendix E.1. Other detailed OUR setups are in Appendix D.1. For class-wise unlearning, each trial selects five random classes as out-of-distribution (OOD) and one target class for unlearning. Using only one OOD class could introduce bias due to class-specific differences. Thus, corresponding experiments are labeled CIFAR20 (-5) and CIFAR100 (-5). MU evaluation metrics follow Section 2.

Privacy Attack Parameters. Class-wise ReA adversaries have limited access to unlearned data, with only 3% for CIFAR20 and 20% for CIFAR100. Each attack experiment consists of 50 trials with randomized class selections, totaling 300 inferred classes. ReA uses an SGD optimizer and an lr set $\{0.001, 0.005, 0.007, 0.01\}$ to perform a multi-lr aggregration strategy. Its convergence threshold is set to 75%. The ablation studies discussing hyperparameters of ReA are presented in Appendix E.3 and Appendix E.4. For sample-wise MIA, following prior work [28], we assume adversaries have a dataset containing training (retained), unlearned, and test data but lack knowledge of their memberships. Here, MIA specifically focuses on the membership of unlearned data rather than training (retained) data. Sample-wise ReA uses an lr of 0.01 for CIFAR10 (SGD) and 5e^{-5} for CIFAR20 (Adam), which is $0.1\times$ the training lr.

Evaluation for MIA follows SOTA ROC analysis [7], measuring MIA performance with TPR at a fixed FPR. Balanced attack accuracy is also reported. Detailed descriptions of OUR hyper-parameters, ReA settings, and platforms are provided in Appendix D.1.

Privacy Attack Benchmarks. To compare with ReA, we evaluate two SOTA attacks: MIA-LiRA [7] and MIA-UP [28], as described in Section 2. Though originally designed for sample-wise MIA, we adapt them to class-wise MIA as follows. Both methods run on the entire set of inferred classes which match the training set size, producing a single positive rate with a fixed threshold τ'. This score serves as the output of $\mathcal{A}'$ in Equation 3. Specifically, MIA-UP replaces shadow models' test samples with OOD classes to serve as the negative samples. MIA-LiRA assumes knowledge of model architecture and hyperparameters, while MIA-UP further requires unlearning algorithm details.

5.2. Evaluation Results in Classification Tasks

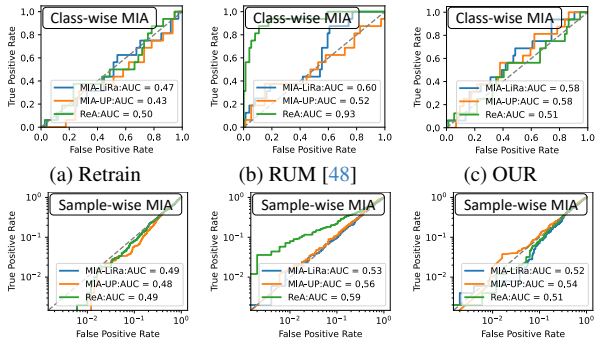
We present class-wise and sample-wise unlearning performance for classification tasks in Tables 1, with subscripts indicating standard deviations and blue values showing differences from the retrained model. The corresponding class-wise and sample-wise privacy attack results for different AMU methods are shown in Table 2 and Table 3, with bold denoting the best results. Table 1 demonstrates that our OUR achieves unlearning efficacy comparable to the ground-truth retrained model on both ViT and ResNet architectures, while exhibiting superior run-time efficiency (RTE) which is lower than 8% of retraining time.

Besides, the privacy vulnerability evaluation reveals two critical insights (Tables 2-3). First, our Reminiscence Attack (ReA) significantly outperforms existing privacy attacks (MIA-LiRA and MIA-UP) across most MU methods. Class-wise MU methods such as random label (RL) [18], and its optimized version Salun [13] and RUM [48], IU [25], and BU [8] exhibit severe vulnerabilities, with ReA achieving over 81% accuracy. This vulnerability likely arises because RL-based methods mainly affect model parameters near the output layer, and IU and BU have limited unlearning efficacy on ViT models. For sample-wise unlearning, IU [25] and FF [17] which lack gradient descent processes, and gradient ascent (GA) [40] show heightened susceptibility to the reminiscence process, with ReA attaining over 55% (on CIFAR10) and 70% (on CIFAR20) accuracy. Second, our OUR method substantially mitigates privacy leakage and reduces class-wise and sample-wise MIA accuracies lower than 54.73% and 53.80%, respectively, which are significantly close to retraining baselines.

ROC Analysis of Privacy Attacks. ROC analysis [7] MIA via the trade-off between true positive rate (TPR) and false positive rate (FPR). The area under the curve (AUC) quantifies attack effectiveness, with higher values indicating better separation between member and non-member distributions. Figures 3 compare ROC curves for MU benchmarks, OUR, and retrained models on CIFAR20 in class-wise MIA and sample-wise MIA, respectively. The results

Table 1. Machine Unlearning (MU) for Classification

Class-wise Unlearning					
MU	TA/%	UA/%	RA/%	MIA Efficacy/%	RTE (s)
CIFAR20 (-5) - ViT					
Retrain	$51.31_{\pm0.11}(0.00)$	$0.00_{\pm0.00}(0.00)$	$99.99_{\pm0.01}(0.00)$	$1.67_{\pm0.10}(0.00)$	1801.26
FT [18]	$49.40_{\pm0.27}(1.91)$	$0.74_{\pm0.58}(0.74)$	$97.89_{\pm0.04}(2.10)$	$1.54_{\pm0.18}(0.13)$	120.99
GA [40]	$49.13_{\pm0.55}(2.18)$	$0.00_{\pm0.00}(0.00)$	$99.01_{\pm0.02}(0.98)$	$92.05_{\pm3.13}(90.38)$	69.46
RL [18]	$50.72_{\pm0.15}(0.59)$	$0.00_{\pm0.00}(0.00)$	$99.98_{\pm0.01}(0.01)$	$1.66_{\pm0.08}(0.00)$	26.04
IU [25]	$48.21_{\pm0.52}(3.10)$	$94.95_{\pm4.46}(94.95)$	$92.21_{\pm0.21}(7.78)$	$20.98_{\pm2.29}(19.31)$	30.32
BU [8]	$50.20_{\pm1.49}(1.11)$	$15.35_{\pm1.23}(15.35)$	$88.73_{\pm0.96}(11.26)$	$15.40_{\pm0.45}(13.73)$	29.92
SCRUB [28]	$51.83_{\pm0.19}(0.52)$	$1.64_{\pm0.14}(1.64)$	$97.54_{\pm0.62}(2.45)$	$5.00_{\pm0.72}(3.33)$	131.58
$l1$ Sparse[25]	$51.34_{\pm0.18}(0.03)$	$0.00_{\pm0.00}(0.00)$	$99.93_{\pm0.02}(0.06)$	$15.40_{\pm0.62}(13.73)$	142.95
SalUn [13]	$51.05_{\pm0.21}(0.28)$	$3.05_{\pm0.15}(3.05)$	$99.96_{\pm0.01}(0.03)$	$1.67_{\pm0.52}(0.00)$	67.41
SFRon [24]	$48.18_{\pm0.25}(3.13)$	$0.00_{\pm0.00}(0.00)$	$95.63_{\pm0.52}(4.36)$	$87.43_{\pm2.35}(85.76)$	95.69
RUM [48]	$51.79_{\pm0.12}(0.46)$	$2.85_{\pm0.28}(2.85)$	$99.88_{\pm0.02}(0.03)$	$0.00_{\pm0.00}(1.67)$	98.28
OUR	$51.98_{\pm0.13}(0.67)$	$0.00_{\pm0.00}(0.00)$	$99.97_{\pm0.01}(0.02)$	$3.67_{\pm0.11}(0.00)$	78.24
CIFAR100 (-5) - ResNet-50					
Retrain	$62.63_{\pm0.14}(0.00)$	$0.00_{\pm0.00}(0.00)$	$99.98_{\pm0.00}(0.00)$	$0.00_{\pm0.00}(0.00)$	2718.51
FT [18]	$63.90_{\pm0.29}(1.27)$	$0.63_{\pm0.31}(0.63)$	$99.98_{\pm0.00}(0.00)$	$0.00_{\pm0.00}(0.00)$	123.46
GA [40]	$61.02_{\pm0.19}(1.61)$	$0.00_{\pm0.00}(0.00)$	$99.96_{\pm0.02}(0.02)$	$14.15_{\pm2.24}(14.15)$	57.48
RL [18]	$63.84_{\pm0.15}(1.21)$	$0.00_{\pm0.08}(0.00)$	$99.98_{\pm0.01}(0.00)$	$6.33_{\pm0.68}(6.33)$	42.58
IU [25]	$61.09_{\pm0.23}(1.54)$	$0.44_{\pm0.05}(0.44)$	$99.95_{\pm0.02}(0.03)$	$0.00_{\pm0.00}(0.00)$	16.76
BU [8]	$61.15_{\pm0.14}(1.48)$	$0.00_{\pm0.00}(0.00)$	$89.55_{\pm3.21}(10.43)$	$0.00_{\pm0.00}(0.00)$	49.15
SCRUB [28]	$61.02_{\pm0.35}(1.61)$	$0.00_{\pm0.00}(0.00)$	$99.97_{\pm0.02}(0.01)$	$0.00_{\pm0.00}(0.00)$	136.81
$l1$ Sparse[25]	$62.13_{\pm0.16}(0.50)$	$0.00_{\pm0.00}(0.00)$	$98.47_{\pm0.14}(1.52)$	$1.33_{\pm0.11}(1.33)$	126.67
SalUn [13]	$62.93_{\pm0.18}(0.30)$	$0.00_{\pm0.00}(0.00)$	$98.90_{\pm0.08}(0.10)$	$6.33_{\pm2.43}(6.33)$	38.29
SFRon [24]	$61.35_{\pm0.12}(0.28)$	$0.00_{\pm0.00}(0.00)$	$95.76_{\pm1.14}(0.04)$	$29.33_{\pm5.47}(29.33)$	150.77
RUM [48]	$62.24_{\pm0.25}(0.39)$	$0.00_{\pm0.00}(0.00)$	$99.07_{\pm0.12}(0.91)$	$0.00_{\pm0.00}(0.00)$	136.44
OUR	$62.88_{\pm0.13}(0.25)$	$0.00_{\pm0.00}(0.00)$	$99.96_{\pm0.02}(0.02)$	$0.00_{\pm0.00}(0.00)$	72.12
10% Random Sample Unlearning (Sample-wise Unlearning)					
MU	TA/%	UA/%	RA/%	MIA Efficacy/%	RTE (s)
CIFAR10 - ResNet-18					
Retrain	$88.98_{\pm0.39}(0.00)$	$88.91_{\pm0.51}(0.00)$	$99.75_{\pm0.21}(0.00)$	$6.40_{\pm0.25}(0.00)$	2401.89
FT [18]	$87.76_{\pm0.26}(1.12)$	$88.36_{\pm0.51}(0.55)$	$97.39_{\pm0.25}(2.36)$	$6.80_{\pm0.31}(0.40)$	145.52
GA [40]	$89.79_{\pm0.26}(0.81)$	$85.73_{\pm2.15}(0.55)$	$98.09_{\pm0.06}(1.66)$	$11.00_{\pm0.37}(4.60)$	116.41
RL [18]	$88.78_{\pm0.28}(0.10)$	$89.39_{\pm0.17}(0.48)$	$96.38_{\pm0.13}(4.37)$	$8.00_{\pm0.42}(1.60)$	43.65
IU [25]	$85.70_{\pm0.21}(3.28)$	$90.02_{\pm0.43}(1.11)$	$91.11_{\pm0.79}(8.64)$	$12.20_{\pm1.50}(5.80)$	30.32
FF [17]	$83.99_{\pm0.19}(4.99)$	$90.39_{\pm0.25}(1.48)$	$90.80_{\pm0.36}(8.85)$	$9.60_{\pm0.14}(3.20)$	2161.07
SCRUB [28]	$89.40_{\pm0.15}(0.42)$	$89.73_{\pm0.21}(0.83)$	$97.19_{\pm0.04}(2.56)$	$8.40_{\pm0.39}(2.00)$	1353.33
$l1$ Sparse[25]	$85.91_{\pm0.21}(4.07)$	$87.03_{\pm0.83}(1.88)$	$94.33_{\pm0.14}(5.42)$	$6.20_{\pm0.08}(0.20)$	140.06
SalUn [13]	$89.89_{\pm0.25}(0.91)$	$89.23_{\pm0.21}(0.32)$	$97.71_{\pm0.08}(2.04)$	$7.20_{\pm0.13}(0.80)$	32.77
SFRon [24]	$90.58_{\pm0.28}(1.60)$	$86.69_{\pm0.79}(2.22)$	$99.02_{\pm0.01}(0.73)$	$13.20_{\pm1.25}(6.80)$	114.39
RUM [48]	$91.16_{\pm0.27}(2.18)$	$88.04_{\pm0.34}(0.87)$	$99.97_{\pm0.01}(0.22)$	$4.20_{\pm0.09}(2.00)$	110.59
OUR	$88.38_{\pm0.19}(0.60)$	$88.99_{\pm0.22}(0.08)$	$99.23_{\pm0.14}(0.52)$	$7.00_{\pm0.31}(0.60)$	91.00
CIFAR20 - ViT					
Retrain	$69.16_{\pm0.24}(0.00)$	$69.07_{\pm0.18}(0.00)$	$99.99_{\pm0.00}(0.00)$	$12.20_{\pm0.35}(0.00)$	2313.38
FT [18]	$64.00_{\pm0.21}(5.16)$	$68.50_{\pm0.25}(0.57)$	$94.29_{\pm0.34}(5.46)$	$13.80_{\pm1.35}(1.60)$	96.42
GA [40]	$61.21_{\pm0.15}(7.95)$	$74.77_{\pm0.23}(5.70)$	$80.66_{\pm2.72}(19.09)$	$24.80_{\pm6.34}(12.60)$	146
RL [18]	$63.69_{\pm0.27}(5.47)$	$67.48_{\pm0.13}(1.59)$	$88.45_{\pm0.31}(11.30)$	$13.80_{\pm1.25}(1.60)$	71.29
IU [25]	$62.21_{\pm0.15}(6.95)$	$82.24_{\pm0.24}(13.17)$	$86.05_{\pm0.76}(13.70)$	$31.40_{\pm3.58}(19.20)$	57.41
FF [17]	$60.19_{\pm0.08}(8.97)$	$88.24_{\pm0.13}(19.17)$	$88.10_{\pm0.45}(11.70)$	$33.24_{\pm4.53}(21.40)$	1680.61
SCRUB [28]	$62.72_{\pm0.21}(6.44)$	$73.51_{\pm1.43}(4.44)$	$76.58_{\pm1.65}(23.17)$	$17.60_{\pm1.32}(5.40)$	194.61
$l1$ Sparse[25]	$62.93_{\pm0.09}(6.23)$	$68.01_{\pm0.11}(1.06)$	$90.49_{\pm0.23}(9.26)$	$14.00_{\pm0.85}(1.80)$	83.30
SalUn [13]	$63.45_{\pm0.27}(5.71)$	$65.79_{\pm0.31}(3.28)$	$92.55_{\pm0.18}(7.20)$	$12.00_{\pm1.75}(0.20)$	133.19
SFRon [24]	$66.20_{\pm0.24}(2.96)$	$69.00_{\pm0.45}(0.07)$	$97.60_{\pm0.08}(2.15)$	$21.80_{\pm3.25}(9.60)$	53.85
RUM [48]	$64.87_{\pm0.18}(4.29)$	$67.13_{\pm0.24}(1.94)$	$91.36_{\pm0.43}(8.39)$	$11.00_{\pm0.76}(1.20)$	103.10
OUR	$67.90_{\pm0.21}(1.92)$	$69.37_{\pm0.19}(0.30)$	$97.72_{\pm0.08}(0.44)$	$12.80_{\pm1.24}(0.60)$	187.56



(a) Retrain (b) RUM [48] (c) OUR

(d) Retrain (e) SCRUB [28] (f) OUR

Figure 3. ROC Analysis of MIA (Selected Methods).

indicate that ReA achieves substantially higher AUC values (up to $1.78\times/1.55\times$) than MIA-LiRA/UP, demonstrating superior membership inference capability. Notably, OUR's curves align closely with retrained models (AUC difference < 0.11), suggesting better privacy preservation. Complete ROC analyses of all AMU appear in Appendix E.5.

5.3. Quantified Analysis of Unlearning Residuals

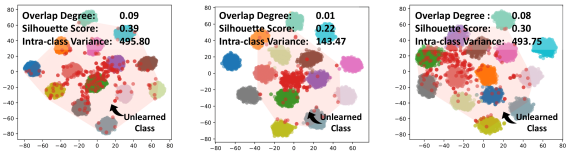
We quantify residuals in MU benchmarks to explain why ReA succeeds, and assess OUR's effectiveness in scrub-

Table 2. Performance of Attacks on Class-wise Unlearning: Inferring the Membership of Unlearned Classes from OOD Classes

Attacks	Balanced Accuracy (TPR @0.1 FPR) / %											
	Retrain	FT	GA	RL	IU	BU	SCRUB	$l1$ Sparse	SalUn	SFRon	RUM	OUR
CIFAR20 (-5)												
MIA-LiRA	52.28(31.25)	59.04(18.75)	61.57(37.50)	50.00(31.25)	64.70(18.75)	52.53(25.00)	50.42 (6.25)	50.00(25.00)	59.29(6.25)	**89.02(87.50)**	61.49(25.00)	**52.45**(6.25)
MIA-UP	50.00(18.75)	56.84(12.50)	64.70(37.50)	61.40(18.75)	69.00(50.00)	64.95(12.50)	50.00(0.00)	50.25(18.75)	67.40(50.00)	73.14(43.75)	50.00(18.75)	50.00(18.75)
ReA	50.00(6.25)	**86.99(75.00)**	**71.03(43.75)**	**95.52(93.75)**	**89.02(87.50)**	**91.05(93.75)**	**68.75(31.25)**	**63.34(25.00)**	**95.27(100.00)**	84.29(81.25)	**85.64(81.25)**	52.28(**12.50**)
CIFAR100 (-5)												
MIA-LiRA	50.00(6.25)	57.52(6.25)	55.49(25.00)	69.00(25.00)	87.67(87.50)	81.42(75.00)	**64.44(31.25)**	59.54(18.75)	69.17(43.75)	65.37(37.50)	66.89(37.50)	50.00(6.50)
MIA-UP	50.08(12.50)	63.60(43.75)	59.54(18.75)	56.59(25.00)	63.09 (25.00)	67.23(37.50)	52.70(6.25)	55.49(25.00)	64.27(37.50)	**73.14(56.25)**	54.05(6.25)	**54.73(12.50)**
ReA	50.00(6.25))	**86.32(75.00)**	**60.22(31.25)**	**82.47(81.50)**	**89.24(89.75)**	**84.17(83.75)**	61.57(31.25)	**62.67(25.00)**	**81.33(62.50)**	65.88(31.25)	**82.77(87.50)**	50.68(0.00)

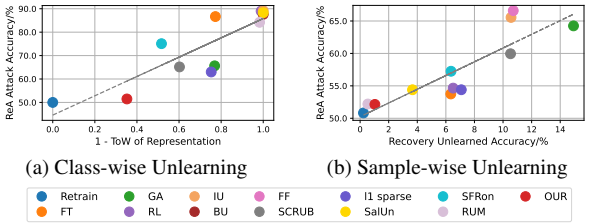
Table 3. Performance of Attacks on Random Sample Forgetting (10%): Infer the Membership of Unlearned Samples

Attacks	Balanced Accuracy (TPR@0.1FPR) / %											
	Retrain	FT	GA	RL	IU	FF	SCRUB	$l1$ Sparse	SalUn[13]	SFRon	RUM	OUR
CIFAR10												
MIA-LiRA	50.63(6.40)	50.60(6.80)	51.50(11.00)	51.50(8.00)	53.60(12.20)	53.60(9.60)	50.90(8.40)	50.90(6.20)	52.40(7.20)	52.20(13.20)	51.00(4.20)	50.69(7.00)
MIA-UP	49.99(9.80)	**50.66(9.92)**	52.72(15.40)	50.00 (**10.24**)	52.84(12.28)	52.50(12.12)	50.34(**9.72**)	50.78(**9.96**)	51.34(**11.44**)	52.68(**14.36**)	**52.82(15.56)**	**50.70(9.88)**
ReA	50.30(7.00)	50.40(8.60)	**55.60(15.60)**	**52.20**(9.60)	**58.40(24.20)**	**57.90(15.00)**	**51.60**(6.00)	**51.20**(6.00)	**53.40**(6.80)	**52.80**(13.60)	51.20(8.20)	50.48(7.20)
CIFAR20												
MIA-LiRA	51.20(12.20)	56.10(13.80)	61.80(24.80)	53.20(**13.80**)	64.80(31.40)	68.20 (33.24)	57.60(17.60)	53.10(**14.00**)	53.80(**12.00**)	56.80(21.80)	52.10(**11.00**)	53.20(**12.80**)
MIA-UP	51.39(9.24)	54.46(10.80)	57.62(12.88)	53.48(8.88)	60.14(12.92)	59.89(12.24)	56.04(11.52)	52.84(9.72)	52.02(8.00)	51.36(20.84)	51.36(8.96)	52.46(9.88)
ReA	51.35(10.80)	**57.00(12.80)**	**72.90(44.00)**	**57.10**(12.20)	**72.70(44.40)**	**75.30(45.20)**	**63.30(33.40)**	**57.60**(12.00)	**55.40**(9.80)	**61.70(28.00)**	**53.20**(10.00)	**53.80**(11.50)



(a) Retrain (b) RUM [48] (c) OUR

Figure 4. Representation space visualization of the unlearned model. OUR's treatment for the *fruit & vegetables* forget class (red) matches retrained models better than RUM. Appendix E.6 provides results for all AMU methods.



(a) Class-wise Unlearning (b) Sample-wise Unlearning

Figure 5. Quantified Residuals versus ReA Attack Accuracy.

bing them. To validate this, we use distinct analyses for class-wise and sample-wise residuals. For class-wise cases, we visualize representation spaces to detect class-specific residuals. For sample-wise cases, we quantify residuals by measuring the recovered unlearned accuracy (UA) by ReA.

Class-wise Residual Visualization. Figure 4 compares t-SNE [42] projections of unlearned models in CIFAR-20 (-5) experiments and quantifies them with representation metrics: (1) **Overlap degree** (o) for class separation, (2) **Silhouette Score** (s) for cluster compactness, and (3) **Intra-class Variance** (var). OUR achieves near-retrained model alignment ($\Delta\text{var} < 3$ vs. 352 for RUM [48]), indicating effective residual scrubbing. Figure 5a further reveals a strong correlation (Pearson $= 0.88$) between representation misalignment (1-ToW(o, s, var)) and ReA accuracy, confirming that representation residuals enable privacy breaches. Full metric definitions and results are in Appendix E.6.

Sample-wise Residual Quantification. For sample-wise cases, we assess residuals by tracking unlearning accuracy recovery during ReA. Figure 5b shows that higher recovered accuracy (ΔUA) correlates with increased privacy leakage (Pearson $= 0.87$). OUR minimizes both recovery ($\Delta\text{UA} = 1.26\%$) and privacy risk (ReA$\leq 53.80\%$), while some methods like GA [40] exhibit strong recovery ($\Delta\text{UA} = 14.85\%$) and high vulnerability (ReA$\leq 72.90\%$).

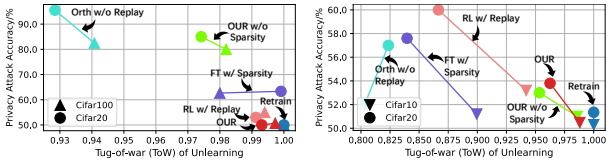
5.4. Analysis of OUR Variants

To validate the necessity of OUR's core components, we evaluate four ablated variants: (1) *Orth w/o Replay* (jointly optimizing orthogonal unlearning and utility preservation), (2) *OUR w/o Sparsity* (removing $l1$ regularization during replay), (3) *RL w/ Replay* (replacing orthogonal unlearning with random label (RL) [18]), and (4) *FT w/ Sparsity* (fine-tuning only with $l1$ regularization [25]).

Figure 6 plots unlearning efficacy (measured by ToW(UA,TA,RA)) against ReA privacy attack accuracy across datasets, and three critical observations emerge. First, the two-phase design is essential. *Orth w/o Replay* performs the worst, which degrades unlearning efficacy by 0.06 and increases ReA attack accuracy to 95.52%, showing that joint optimization in unlearning weakens both objectives. Second, orthogonal unlearning proves superior to alternatives. OUR reduces ReA accuracy by $2.89 - 6.12\%$ compared to *RL w/ Replay*, as it removes more internal residuals. Third, $l1$ regularization in OUR enhances class-wise unlearning (reducing ReA accuracy by 35.07%), yet has a limited effect on sample-wise cases where the separated replay phase dominates. The synergy of orthogonal unlearning, two-phase design and $l1$ regularization achieves both effective unlearning and privacy protection.

5.5. Experiments on Image Generation

We evaluate privacy risks in identity-sensitive face generation using SD models. The target model synthesizes two identities (e.g., "a photo of Laura") with distinct features. Concept unlearning is performed to erase a target identity. Its unlearning efficacy is measured by Unlearning Identity Score Matching (ISM) [43], abbreviated as **UISM** Retrained generation quality is assessed using three metrics:



(a) Class-wise Unlearning (b) Sample-wise Unlearning

Figure 6. Unlearning performance of OUR variants versus their privacy vulnerability test by ReA.

Table 4. Performance of Identity-specific Concept Unlearning

Metrics	Retrain	SalUn [13]	Meta-Un [15]	OUR	SalUn-OUR
UISM	$0.13_{\pm 0.02}$(0.00)	$0.49_{\pm 0.03}$(0.36)	$0.37_{\pm 0.05}$(0.24)	$0.16_{\pm 0.02}$(0.03)	$0.26_{\pm 0.01}$(0.13)
RISM	$0.76_{\pm 0.02}$(0.00)	$0.71_{\pm 0.02}$(0.05)	$0.74_{\pm 0.04}$(0.02)	$0.68_{\pm 0.01}$(0.08)	$0.74_{\pm 0.02}$(0.02)
FID(↓)	$101.82_{\pm 8.33}$	$108.23_{\pm 7.25}$	$122.76_{\pm 11.35}$	$110.31_{\pm 7.69}$	$103.55_{\pm 11.35}$
BRISQUE(↓)	$10.76_{\pm 1.38}$	$15.15_{\pm 0.41}$	$15.02_{\pm 2.03}$	$15.58_{\pm 3.90}$	$12.03_{\pm 3.82}$
RTE (s)	937.50	239.31	485.31	110.25	113.83

Table 5. Images of Identity-specific Concept Unlearning

Target	Retrain	SalUn [13]	Meta-Un [15]	OUR	SalUn-OUR
Unlearned Identity. Prompt: "*a photo of Sarah*"					
Retained Identity. Prompt: "*a photo of Laura*"					

Retaining ISM (**RISM**) for identity accuracy, **FID** [13] for distribution fidelity, and **BRISQUE** [32] for artifacts. Appendix E.7 details setups and metrics and evaluates more unlearning benchmarks, e.g., SFRon [24].

As shown in Table 4, OUR achieves near-retrained UISM with 12% of retraining time but initially causes RISM degradation by 0.08. To mitigate this, we integrate SalUn's parameter freezing strategy to preserve retained identity features, improving RISM by 0.06 without compromising running-time efficiency. Visual results in Table 5 confirm that (SalUn-)OUR effectively removes target identities while maintaining retained output quality.

We design two MIA to evaluate MU's privacy leakage and report the results in Table 6: (1) **ReA**, adapted from class-wise ReA, and (2) **DiffAtk** [47], which employs adversarial prompts to revive forgotten identities within limited iterations. To be practical, attackers are assumed to have access to public photos of the forgotten identity (not original training data) and lack knowledge of the forgotten prompt. Both attacks run for 100 iterations, with success measured by the resonance index (Idx_r) where ISM exceeds 0.70. ReA proves most effective, requiring 25 fewer Idx_r in SalUn, while DiffAtk consistently fails. Second, orthogonal unlearning in OUR enhanced security for portrait generation compared to existing unlearning methods. OUR achieves a resilience index (Idx_r) of 70 (near-retrained models), even better than Meta-unlearning (Meta-Un) [15], which is designed to prevent forgotten concept recovery.

Table 6. Privacy Attack against Concept Unlearning

Attacks	Resonance Index, Idx_r, (Maximum = 10) × 10 @ ISM				
	Retrain	SalUn [13]	Meta-Un [15]	OUR	SalUn-OUR
DiffAtk[47]	$10_{\pm 0.0}$@0.18	$10_{\pm 0.0}$@0.10	$10_{\pm 0.0}$@0.09	$10_{\pm 0.0}$@0.10	$10_{\pm 0.0}$@0.08
ReA	$6.5_{\pm 1.0}$@0.71	$2_{\pm 0.5}$@0.77	$4.5_{\pm 1}$@0.71	$7.5_{\pm 1.0}$@0.70	$7_{\pm 1}$@0.72

6. Related Works

Privacy Hazards in Machine Unlearning. The growing adoption of machine unlearning (MU) has raised concerns about its unintended privacy risks. Current research identifies two primary leakage sources: (1) updates between model versions before and after unlearning [5, 9, 22], and (2) knowledge residuals caused by approximate unlearning algorithms [28]. While version control addresses the first, the second reveals more profound flaws: imperfect alignment between unlearned and test data creates measurable discrepancies. Crucially, we reveal that these discrepancies are not merely superficial; they manifest as residuals in the loss landscape. By exposing these hidden residuals, our work shows that existing unlearning methods inadvertently leak privacy. This highlights the need for stricter privacy guarantees beyond output-level evaluations.

Forgotten Knowledge Recovery Phenomenon. Recent studies [15, 23, 35] on safety-aligned generative models have observed the *Forgotten Knowledge Recovery Phenomenon* (FRP), where supposedly removed harmful knowledge unexpectedly re-emerges through targeted relearning. This suggests that unlearned targets leave residual traces. Existing work [15, 23] attributes these residuals to semantic/knowledge overlaps between harmful and other concepts. We go beyond this insights and examine FRP across a broad range of approximate machine unlearning (MU) methods [24, 25, 28, 48] in closed-problem settings (with well-defined input/output spaces), and leverage FRP to uncover privacy risks. We find that **algorithm-induced residuals persist regardless of semantic correlations**, which poses measurable membership privacy threats.

7. Conclusion

This paper reveals pervasive privacy vulnerabilities in approximate unlearning algorithms induced by implicit residuals. To exploit these residuals, we introduce Reminiscence Attack (ReA), a novel membership inference attack (MIA) that compromises unlearned data privacy. ReA extends its targets beyond unlearned samples to unlearned classes and concepts, posing a broader privacy threat. Compared to existing attacks like MIA-UP [28] relying on output-space residuals, ReA achieves up to 1.90× higher attack accuracy. To mitigate this risk, we propose Orthogonal Unlearning & Replay (OUR), a framework that systematically scrubbing residuals to reduce privacy leakage. OUR generalizes across class-wise, sample-wise, and concept unlearning, achieving near-retrained unlearning efficacy while lowering privacy attack accuracy to near-random levels.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant No: 92270123, 62372122, and 62372130), and the Research Grants Council, Hong Kong SAR, China (Grant No: 15210023, 15224124, 25207224).

References

[1] Bill text. https://www.priv.gc.ca/en/privacy-topics/privacy-laws-in-canada/. 1

[2] Gordon Anderson, Oliver Linton, and Yoon-Jae Whang. Nonparametric estimation and inference about the overlap of two distributions. *Journal of Econometrics*, 171(1):1–23, 2012. 15

[3] Li Bai, Haibo Hu, Qingqing Ye, Haoyang Li, Leixia Wang, and Jianliang Xu. Membership inference attacks and defenses in federated learning: A survey. *ACM Comput. Surv.*, 57(4), 2025. 3

[4] Li Bai, Haibo Hu, Qingqing Ye, Jianliang Xu, Jin Li, Chengfang Fang, and Jie Shi. Rmr: A relative membership risk measure for machine learning models. *IEEE Transactions on Dependable and Secure Computing*, 2025. 3

[5] Martin Bertran, Shuai Tang, Michael Kearns, Jamie H Morgenstern, Aaron Roth, and Steven Z Wu. Reconstruction attacks on machine unlearning: Simple models are vulnerable. *Advances in Neural Information Processing Systems*, 37:104995–105016, 2025. 2, 3, 8

[6] Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *42nd IEEE Symposium on Security and Privacy, SP 2021, San Francisco, CA, USA, 24-27 May 2021*, pages 141–159. IEEE, 2021. 1

[7] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914. IEEE, 2022. 2, 3, 4, 5, 6

[8] Min Chen, Weizhuo Gao, Gaoyang Liu, Kai Peng, and Chen Wang. Boundary unlearning: Rapid forgetting of deep networks via shifting the decision boundary. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 7766–7775. IEEE, 2023. 6, 14, 17

[9] Min Chen, Zhikun Zhang, Tianhao Wang, Michael Backes, Mathias Humbert, and Yang Zhang. When machine unlearning jeopardizes privacy. In Yongdae Kim, Jong Kim, Giovanni Vigna, and Elaine Shi, editors, *CCS '21: 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, Republic of Korea, November 15 - 19, 2021*, pages 896–911. ACM, 2021. 2, 3, 8

[10] Vikram S. Chundawat, Ayush K. Tarun, Murari Mandal, and Mohan S. Kankanhalli. Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher. In Brian Williams, Yiling Chen, and Jennifer Neville, editors, *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 7210–7217. AAAI Press, 2023. 1, 13

[11] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 16

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 5

[13] Chongyu Fan, Jiancheng Liu, Yihua Zhang, Dennis Wei, Eric Wong, and Sijia Liu. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *International Conference on Learning Representations*, 2024. 2, 6, 7, 8, 14, 16, 17

[14] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1322–1333, 2015.

[15] Hongcheng Gao, Tianyu Pang, Chao Du, Taihang Hu, Zhijie Deng, and Min Lin. Meta-unlearning on diffusion models: Preventing relearning unlearned concepts. *CoRR*, abs/2410.12777, 2024. 2, 8, 16

[16] Lei Gao, Yue Niu, Tingting Tang, Salman Avestimehr, and Murali Annavaram. Ethos: Rectifying language models in orthogonal parameter space. In *NAACL-HLT (Findings)*, 2024. 5

[17] Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19,*

2020, pages 9301–9309. Computer Vision Foundation / IEEE, 2020. 1, 2, 6, 13

[18] Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11516–11524, 2021. 4, 5, 6, 7, 13, 17

[19] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009. 15

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. 5

[21] Tuan Hoang, Santu Rana, Sunil Gupta, and Svetha Venkatesh. Learn to unlearn for deep neural networks: Minimizing unlearning interference with gradient projection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4819–4828, 2024. 5

[22] Hongsheng Hu, Shuo Wang, Tian Dong, and Minhui Xue. Learn what you want to unlearn: Unlearning inversion attacks against machine unlearning. In *IEEE Symposium on Security and Privacy, SP 2024, San Francisco, CA, USA, May 19-23, 2024*, pages 3257–3275. IEEE, 2024. 2, 3, 8

[23] Shengyuan Hu, Yiwei Fu, Zhiwei Steven Wu, and Virginia Smith. Jogging the memory of unlearned llms through targeted relearning attacks. *arXiv preprint arXiv:2406.13356*, 2024. 8

[24] Zhehao Huang, Xinwen Cheng, JingHao Zheng, Haoran Wang, Zhengbao He, Tao Li, and Xiaolin Huang. Unified gradient-based machine unlearning with remain geometry enhancement. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. 1, 6, 8, 14, 17

[25] Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. Model sparsity can simplify machine unlearning. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 1, 2, 3, 5, 6, 7, 8, 13, 14, 17

[26] Krish Kabra and Guha Balakrishnan. F? d: On understanding the role of deep feature spaces on face generation evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8327–8332, 2024. 17

[27] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 5

[28] Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. *Advances in neural information processing systems*, 36:1957–1987, 2023. 2, 3, 5, 6, 8, 14, 17

[29] Haoyang Li, Li Bai, Qingqing Ye, Haibo Hu, Yaxin Xiao, Huadi Zheng, and Jianliang Xu. A sample-level evaluation and generative framework for model inversion attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 18287–18295, 2025. 1

[30] Zitong Li, Qingqing Ye, and Haibo Hu. FUNU: boosting machine unlearning efficiency by filtering unnecessary unlearning. In Guodong Long, Michale Blumestein, Yi Chang, Liane Lewin-Eytan, Zi Helen Huang, and Elad Yom-Tov, editors, *Proceedings of the ACM on Web Conference 2025, WWW 2025, Sydney, NSW, Australia, 28 April 2025- 2 May 2025*, pages 3366–3376. ACM, 2025. 3

[31] Peizhuo Lv, Chang Yue, Ruigang Liang, Yunfei Yang, Shengzhi Zhang, Hualong Ma, and Kai Chen. A data-free backdoor injection approach in neural networks. In Joseph A. Calandrino and Carmela Troncoso, editors, *32nd USENIX Security Symposium, USENIX Security 2023, Anaheim, CA, USA, August 9-11, 2023*, pages 2671–2688. USENIX Association, 2023. 4

[32] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 8

[33] Alessandro Mantelero. The eu proposal for a general data protection regulation and the roots of the 'right to be forgotten'. *Computer Law & Security Review*, 29(3):229–235, 2013. 1

[34] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Trans. Image Process.*, 21(12):4695–4708, 2012. 17

[35] Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-

tuning aligned language models compromises safety, even when users do not intend to! In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 8

[36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE, 2022. 2, 5, 16

[37] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 22500–22510. IEEE, 2023. 5

[38] Ketan Rajshekhar Shahapure and Charles Nicholas. Cluster quality analysis using silhouette score. In *2020 IEEE 7th international conference on data science and advanced analytics (DSAA)*, pages 747–748. IEEE, 2020. 15

[39] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 3–18. IEEE Computer Society, 2017. 1, 3

[40] Anvith Thudi, Gabriel Deza, Varun Chandrasekaran, and Nicolas Papernot. Unrolling sgd: Understanding factors influencing machine unlearning. In *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*, pages 303–319. IEEE, 2022. 6, 7, 13, 14

[41] Jean-Baptiste Truong, Pratyush Maini, Robert J Walls, and Nicolas Papernot. Data-free model extraction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4771–4780, 2021. 4, 18

[42] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 7

[43] Thanh Van Le, Hao Phung, Thuan Hoang Nguyen, Quan Dao, Ngoc N Tran, and Anh Tran. Anti-dreambooth: Protecting users from personalized text-to-image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2116–2127, 2023. 7, 16

[44] Yuan Wang, Ouxiang Li, Tingting Mu, Yanbin Hao, Kuien Liu, Xiang Wang, and Xiangnan He. Precise, fast, and low-cost concept erasure in value space: Orthogonal complement matters. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 28759–28768, 2025. 5

[45] Alexander Warnecke, Lukas Pirch, Christian Wressnegger, and Konrad Rieck. Machine unlearning of features and labels. In *30th Annual Network and Distributed System Security Symposium, NDSS 2023, San Diego, California, USA, February 27 - March 3, 2023*. The Internet Society, 2023. 1, 3, 13, 17

[46] Yaxin Xiao, Qingqing Ye, Haibo Hu, Huadi Zheng, Chengfang Fang, and Jie Shi. Mexmi: Pool-based active model extraction crossover membership inference. In *Advances in Neural Information Processing Systems*, 2022. 1

[47] Yimeng Zhang, Jinghan Jia, Xin Chen, Aochuan Chen, Yihua Zhang, Jiancheng Liu, Ke Ding, and Sijia Liu. To generate or not? safety-driven unlearned diffusion models are still easy to generate unsafe images... for now. In *European Conference on Computer Vision*, pages 385–403. Springer, 2024. 8, 16, 17

[48] Kairan Zhao, Meghdad Kurmanji, George-Octavian Barbulescu, Eleni Triantafillou, and Peter Triantafillou. What makes unlearning hard and what to do about it. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. 1, 3, 4, 6, 7, 8, 13, 14, 17