Forgetting Knowledge Localization and Isolation for Continual Forgetting of Pre-trained Vision Models

Zhiwen Yang[1, 3*], Jiehua Zhang[2], Chenggang Yan[1], Yuhan Gao[1], Zongpeng Li[1], Xichun Sheng[4], Liang Li[3†]

[1]Hangzhou Dianzi University
[2]Xi'an Jiaotong University
[3]Institute of Computing Technology, Chinese Academy of Sciences
[4]Macao Polytechnic University
zhiwen.yang@hdu.edu.cn, jiehua.zhang@stu.xjtu.edu.cn, cgyan@hdu.edu.cn, yuhangao@hdu.edu.cn, zongpeng@hdu.edu.cn, p2314922@mpu.edu.mo, liang.li@ict.ac.cn

Abstract

Continual forgetting task aims to continuously remove multiple target knowledge subsets from pre-trained models while maintaining the integrity of remaining knowledge. Existing methods suffer from both incomplete forgetting of target knowledge and unintended forgetting of indistinguishable remaining knowledge. To address these challenges, we propose the forgetting knowledge localization and isolation for continual forgetting in pre-trained vision models which precisely forgets target knowledge while reducing over-forgetting of remaining knowledge. To achieve precise forgetting, we first propose the forgetting knowledge layer localization to explore layers in the model which are more related to forgetting knowledge. Then, we design the forgetting knowledge parameter isolation to isolate the parameters sensitive to forgetting knowledge in these selected layers, mitigating over-forgetting of remaining knowledge. Finally, we fine-tune these isolated parameters and freeze the remaining parameters to achieve efficient forgetting while maintaining high performance on retained datasets. Extensive experimental results demonstrate that our method achieves superior performance over state-of-the-art methods across multiple continual forgetting tasks.

Introduction

Pre-trained vision models achieve excellent performance on massive datasets, but these datasets often contain erroneous and privacy-sensitive content, potentially causing issues like racial discrimination and gender bias (Crawford and Paglen 2021; Paglen 2020). Meanwhile, growing privacy awareness and regulatory requirements (GDPR 2018) have increased data deletion requests, which naturally occur as continuous sequences in practice. However, traditional machine unlearning methods focus on single-instance deletion and cannot effectively handle continual deletion scenarios. This motivates the continual forgetting task (Zhao et al. 2024), which requires continuously forgetting multiple target knowledge subsets from pre-trained models while preserving remaining knowledge integrity (Figure 1(a)).

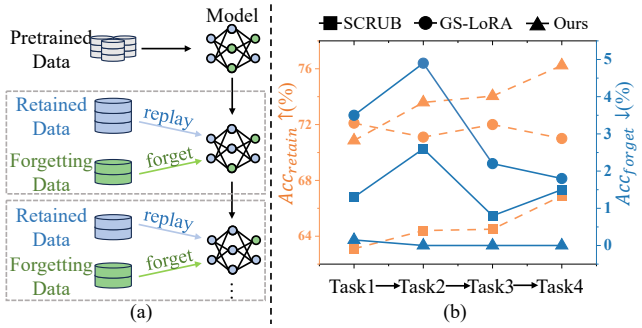

Figure 1: (a) The paradigm of continual forgetting task. (b) Comparison of our method with the state-of-the-art continual forgetting method GS-LoRA and machine unlearning method SCRUB. We conduct 4 forgetting tasks which forget 20 classes in each task. Our method achieves better performance on both forgetting and retained datasets.

The primary challenges of continual forgetting task lie in how to precisely forget specific knowledge from the model while preserving the remaining knowledge. Unlike machine unlearning task, continual forgetting faces the persistent damage to remaining knowledge caused by continuous specific knowledge forgetting, making it a more challenging task. To address these challenges, a direct strategy is to retrain a model using the remaining data whenever there is a deletion request. While this approach can obtain excellent performance, it requires substantial computational resources and time. Therefore, researchers focus on applying machine unlearning methods for pre-trained models to achieve approximate forgetting, thereby alleviating the resource consumption issue associated with retraining. However, traditional machine unlearning methods such as gradient ascent (Jia et al. 2024) and representation misdirection (Huu-Tien et al. 2025) are prone to model collapse or incomplete forgetting. Moreover, most methods (Sekhari et al. 2021; Golatkar, Achille, and Soatto 2020; Kurmanji et al. 2023) are optimized only for single forgetting tasks, leading to severe error accumulation after multiple forgetting tasks, rendering models unable to maintain stable performance on

*This work is done during the intern in VIPL group, ICT, CAS.
†Corresponding Author

the retained dataset as shown in Figure 1(b). The pioneering continual forgetting method GS-LoRA (Zhao et al. 2024) achieves promising performance by applying group sparsity loss to fine-tune pre-trained models with LoRA (Hu et al. 2022). However, since it performs LoRA fine-tuning on all feedforward layers in the model (where knowledge for both retain and forget sets is stored in these weights), it inevitably affects the remaining knowledge in the model.

To address above issues, we propose the forgetting knowledge localization and isolation method to handle continual forgetting of pre-trained vision models. Our method transforms knowledge forgetting from a global optimization challenge into a localized intervention strategy which achieves precisely forgetting target knowledge while mitigating over-forgetting of remaining knowledge. Specifically, we propose the forgetting knowledge layer localization module (FKLL) to adaptively explore network layers that are more related to forgetting knowledge. First, we analyze the forgetting knowledge sensitivity distribution across layers based on our designed forgetting knowledge sensitive ratio metric, which assesses the relative sensitivity of each layer with respect to knowledge forgetting and knowledge retention. Then we explore layers that are more sensitive to forgetting knowledge while being insensitive to remaining knowledge. Next, we propose the forgetting knowledge parameter isolation module (FKPI) to decouple parameters within these selected layers based on their sensitivity to feature differences between forgetting and remaining knowledge. Parameters sensitive to these differences are identified as containing task-specific information critical for encoding the forgetting knowledge and are subsequently fine-tuned using learnable LoRA to remove forgetting knowledge. Conversely, parameters that demonstrate low sensitivity to the feature differences are recognized as containing task-shared knowledge that is essential for preserving the model's general capabilities and remaining knowledge. These insensitive parameters are frozen during training to prevent over-forgetting of remaining knowledge.

Furthermore, we employ a forgetting knowledge isolation loss to mitigate feature drift phenomena (Shan et al. 2024) during knowledge forgetting that could lead to performance degradation on the retained dataset. Meanwhile, we introduce a retained knowledge alignment loss to mitigate the impact of continuous forgetting tasks on remaining knowledge and fully leverage the representational capabilities of the pre-trained model on the retained dataset. Based on these strategies, our method achieves efficient continual forgetting while maintaining high performance on the retained dataset across multiple datasets.

Our main contribution can be summarized as follows:

- We propose forgetting knowledge layer localization module to adaptively locate network layers most closely associated with forgetting knowledge which achieves precisely forgetting the target knowledge.
- We propose forgetting knowledge parameter isolation module to isolate and fine-tune parameters sensitive to forgetting knowledge in the selected layers, mitigating over-forgetting on remaining knowledge.
- We conduct extensive experiments on various continual forgetting tasks, demonstrating that our method achieves state-of-the-art performance on both forgetting and retained datasets.

Related Work

Machine Unlearning

The advancement of deep learning in visual and multimodal tasks (Li et al. 2025; Zhang et al. 2024a; Tu et al. 2024; Liu et al. 2022; Li et al. 2022; Zheng et al. 2022, 2023; Lu et al. 2024; Wang et al. 2024) has led to a growing focus on model safety, thereby fueling growth in emerging fields like machine unlearning. Machine unlearning aims to eliminate the influence of specific data from pre-trained models to realize the "right to be forgotten" (GDPR 2018) and enhance AI safety (Bourtoule et al. 2021; Brophy and Lowd 2021; Shibata et al. 2021; Xu et al. 2024; Li et al. 2023; Liu et al. 2024b). While retraining from scratch provides an effective solution, it incurs prohibitive computational costs for large-scale models (Baumhauer, Schöttle, and Zeppelzauer 2022; Brophy and Lowd 2021). Early approaches focus on traditional machine learning algorithms such as linear regression, k-means clustering, and support vector machines (Chen et al. 2019; Izzo et al. 2021; Mahadevan and Mathioudakis 2021; Sun et al. 2023), leveraging convexity or theoretical constraints that are not directly applicable to deep learning models. Moreover, the distributed and entangled nature of learned representations makes selective knowledge removal particularly challenging, as it often causes catastrophic interference with remaining knowledge. Despite growing interest, achieving efficient and stable unlearning remains fundamentally challenging, especially in continuous deletion scenarios.

Continual Forgetting

Continual forgetting requires the model to continuously forget target knowledge while retaining the remaining knowledge (Zhao et al. 2024). It is a more challenging task than machine unlearning, as it requires continuously protecting the remaining knowledge from the influence of forgetting tasks. Different from continual learning, which aims to continuously learn new knowledge while retaining the old knowledge (Zhang et al. 2024b; Lin et al. 2024; Zhai et al. 2024; Cao et al. 2024; Liu et al. 2024a), continual forgetting has an oppsite optimization goal. Previous work focus on addressing concept erasure problems in generative models (Thakral et al. 2025) or large language models (Gao et al. 2024), and these methods cannot be directly applied to visual pre-trained models. The pioneering work GS-LoRA (Zhao et al. 2024) achieves promising performance by applying group sparsity loss to fine-tune pre-trained models with LoRA. However, since it fine-tune all FFN layers in the model, it inevitably affects the knowledge of retained dataset in the model. Our method addresses this issue by isolating and fine-tuning parameters sensitive to forgetting knowledge while freezing parameters insensitive to forgetting knowledge, thereby mitigating over-forgetting on remaining knowledge.

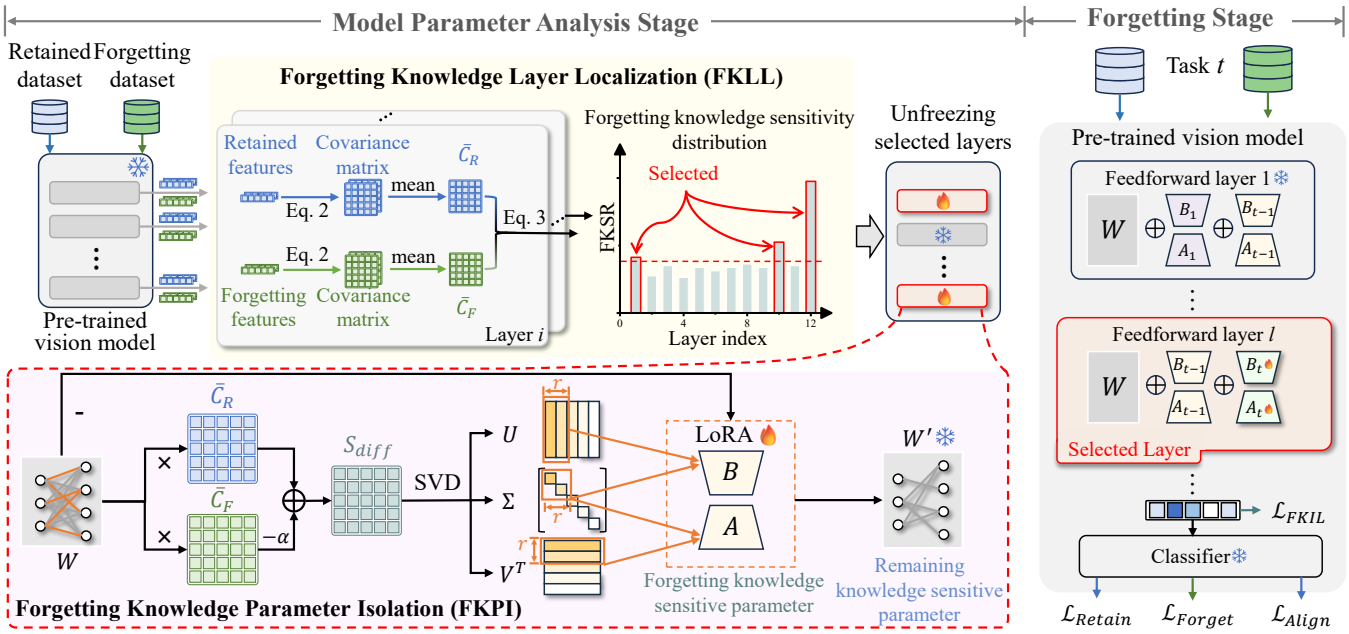

Figure 2: The overall pipeline of our method. We propose the FKLL and FKPI to select the most suitable layers and parameters for forgetting tasks. Finally, we employ a forgetting knowledge isolation loss to mitigate feature drift during the forgetting process and introduce a retained knowledge alignment loss to prevent the over-forgetting of remaining knowledge.

Method

Problem Formulation

Continual forgetting task aims to continuously forget specific knowledge subsets from a pre-trained model while retaining the remaining knowledge. Formally, given a pre-trained model f_θ trained on the dataset D and a sequence of forgetting requests $\mathcal{F} = \{F_1, F_2, \ldots, F_n, \ldots\}$, where each F_n represents a set of classes to be forgetting corresponding the dataset $D_f = \{D_{f_n}\}$ containing knowledge to be forgetting, the goal is to update the model parameters θ such that the model can effectively forget the knowledge associated with F_n while maintaining performance on the retained dataset D_{r_n}. Same as the pioneering work GS-LoRA (Zhao et al. 2024), we assume that the forgetting requests are independent and $|D_r| + |D_f| \ll |D|$. After each forgetting request F_n, the model is updated to a new state f_{θ_n} such that:

$$f_{\theta_n} : \mathcal{X}_{D_{f_n}} \xrightarrow{\theta_n} \mathcal{Y}_{D_{f_n}}, \mathcal{X}_{D_{r_n}} \xrightarrow{\theta_n} \mathcal{Y}_{D_{r_n}}, \tag{1}$$

where $\mathcal{X}_{D_{f_n}}$ and $\mathcal{Y}_{D_{f_n}}$ are the input and output of the forgetting dataset D_{f_n}, and $\mathcal{X}_{D_{r_n}}$ and $\mathcal{Y}_{D_{r_n}}$ are the input and output of the retained dataset D_{r_n}. The goal is to minimize the forgetting loss on the forgetting dataset while maximizing the performance on the retained dataset.

Model Parameter Analysis

To achieve precise and effective forgetting of targeted knowledge while minimizing the impact of forgetting tasks on remaining knowledge, we identify parameters in the model that are more sensitive to forgetting knowledge as shown in the Figure 2. We categorize model parameters into two types: those sensitive to forgetting knowledge and those sensitive to remaining knowledge. For the former, we fine-tune them to achieve forgetting, while for the latter, we freeze them to maintain the integrity of remaining knowledge. Therefore, we adopt a progressively refined approach from layers to intra-layer parameters to localize these two types of parameters. For convinence, we omit the subscript n in the following sections, which represents the forgetting request index, and focus on the current forgetting request.

Forgetting Knowledge Layer Localization. To precisely locate network layers sensitive to forgetting knowledge, we first analyze the sensitivity of each network layer to forgetting and remaining knowledge. We use feature covariance to measure the sensitivity of each network layer to these two types of knowledge. Specifically, we compute the covariance matrices of the input features for each network layer on the forgetting and retained datasets:

$$C_R^l = \frac{1}{N_r - 1} \sum_{i=1}^{N_r} (x_i - \mu_r)(x_i - \mu_r)^T, \tag{2}$$

$$C_F^l = \frac{1}{N_f - 1} \sum_{i=1}^{N_f} (x_i - \mu_f)(x_i - \mu_f)^T, \tag{3}$$

where C_F^l and C_R^l are the covariance matrices for the forgetting and retained datasets for the l-th layer respectively; N_f and N_r are the number of samples in the forgetting and retained datasets, respectively; x_i is the input feature of the i-th sample, and μ_f and μ_r are the mean features for the forgetting and retained datasets, respectively. Then we compute the mean covariance matrix for each layer by averaging

the covariance matrices of all samples in the forgetting and retained datasets:

$$\bar{C}_R = \frac{1}{N_l}\sum_{l=1}^{N_l} C_R^l, \bar{C}_F = \frac{1}{N_l}\sum_{l=1}^{N_l} C_F^l, \tag{4}$$

where N_l is the number of layers in the model. Furthermore, we define a Forgetting Knowledge Sensitive Ratio (FKSR) metric to quantify the sensitivity of each layer to forgetting knowledge. We project covariance with the weight matrix, compute its trace to quantify the overall shift, and apply regularization to avoid extreme values:

$$FKSR = \frac{\mathrm{Tr}\left(W\bar{C}_F W^\top\right)}{\mathrm{Tr}\left(W\bar{C}_R W^\top\right)} \cdot \frac{\left\|\bar{C}_R\right\|}{\left\|\bar{C}_F\right\|}, \tag{5}$$

where W is the weight matrix of the layer, $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix, and $\|\cdot\|$ denotes the Frobenius norm. A higher FKSR indicates that the layer is more sensitive to forgetting knowledge. We select top-K layers with high FKSR values as forgetting-sensitive layers for further processing. The selected layers are denoted as:

$$Layers_{selected} = \{l_1, l_2, \ldots, l_K\}, \tag{6}$$

where l_k is the k-th selected layer and K is a hyperparameter that determines the number of layers to be selected.

Forgetting Knowledge Parameter Isolation. To further isolate parameters within these selected forgetting knowledge-sensitive layers, we analyze the sensitivity of parameters within these layers to the difference between forgetting and remaining knowledge. Specifically, we first compute the difference sensitive matrix as follows:

$$S_{diff} = W\bar{C}_R - \alpha W\bar{C}_F, \tag{7}$$

where α is a hyperparameter that controls the balance between forgetting and remaining knowledge. Then, we perform singular value decomposition (SVD) on the difference sensitive matrix S_{diff} to obtain the low-rank approximation:

$$\mathrm{SVD}\left(S_{diff}\right) = U\Sigma V^\top = \sum_{i=1}^{R} \sigma_i \mathbf{u}_i \mathbf{v}_i^\top, \tag{8}$$

where U and V are the left and right singular vectors, Σ is the diagonal matrix of singular values, and R is the total number of singular values of S_{diff}. We then select the top-r singular values and their corresponding singular vectors to form the low-rank matrices:

$$B = U_{[:,:r]}\sqrt{\Sigma}_{[:r]} \tag{9}$$

$$A = \sqrt{\Sigma}_{[:r]}\left(V^\top S_{diff}^{-1}\right)_{[:r,:]}, \tag{10}$$

where B and A are the low-rank matrices of LoRA, and r is a hyperparameter that determines the rank of the approximation. Finally, we construct the forgetting knowledge parameter isolation module as:

$$W' = W - BA, \tag{11}$$

where W' is the updated weight matrix after isolating the forgetting knowledge parameters. The parameters in B and A are then treated as learnable parameters, while the original weights W are frozen. This allows us to fine-tune only the forgetting knowledge-sensitive parameters while preserving the remaining knowledge in the original weights.

Loss Function

In this section, we introduce the loss function for the continual forgetting task, which consists of four main components: selective forgetting loss, knowledge retention loss, forgetting knowledge isolation loss and retained knowledge alignment loss.

Selective Forgetting Loss

Same as GS-LoRA (Zhao et al. 2024), we use a selective forgetting loss to unlearn the forgetting knowledge. The selective forgetting loss is defined as:

$$\mathcal{L}_{\text{Forget}} = \mathrm{ReLU}\left(\mathrm{BND} - \mathcal{L}\left(f_{M_{t-1}}\left(\mathcal{X}_{f_t}\right), \mathcal{Y}_{f_t}\right)\right), \tag{12}$$

where $\mathcal{L}$ is the cross-entropy loss, $f_{M_{t-1}}$ is the model before the forgetting request, $\mathcal{X}_{f_t}$ and $\mathcal{Y}_{f_t}$ are the input and label of the forgetting dataset respectively, BND is a hyperparameter that controls the boundary for forgetting.

Knowledge Retention Loss

To maintain the performance on the retained dataset, we maintain a small replay buffer containing a subset of the retained dataset, denoted as $\mathcal{X}_{r_t}$ and $\mathcal{Y}_{r_t}$. We use a knowledge retention loss to allevate the forgetting of remaining knowledge. The knowledge retention loss is defined as:

$$\mathcal{L}_{\text{Retain}} = \mathcal{L}\left(f_{M_{t-1}}\left(\mathcal{X}_{r_t}\right), \mathcal{Y}_{r_t}\right). \tag{13}$$

The knowledge retention loss encourages the model to maintain performance on the retained dataset.

Forgetting Knowledge Isolation Loss. We introduce a forgetting knowledge isolation loss to encourage the forgetting knowledge parameters to be orthogonal to the remaining knowledge parameters. The forgetting knowledge isolation loss is defined as:

$$\mathcal{L}_{FKIL} = \frac{1}{N}\sum_{i=1}^{N} \max\left(d\left(a_i, p_i\right) - d\left(a_i, n_i\right) + margin, 0\right), \tag{14}$$

where $d(\cdot,\cdot)$ is the distance function, a_i is the feature of anchor sample from the forgetting dataset, p_i is feature of the positive sample from the forgetting dataset, and n_i is feature of the negative sample from the retained dataset. $\mathcal{L}_{FKIL}$ encourages the parameters contain forgetting knowledge to be orthogonal to the parameters contain remaining knowledge. Only the samples from the forgetting set are utilized as anchor points. In practice, we use 1 minus cosine similarity as the distance function and set the margin to 1.

Retained Knowledge Alignment Loss. To fully leverage the representational capabilities of the pre-trained model on the retained dataset, we align the output logits of the model from the previous forgetting stage with those of the current forgetting stage. The retained knowledge alignment loss is defined as:

$$\mathcal{L}_{\text{Align}} = \mathcal{L}\left(f_{M_{t-1}}\left(\mathcal{X}_{r_t}\right), f_{M_t}\left(\mathcal{X}_{r_t}\right)\right). \tag{15}$$

$\mathcal{L}_{\text{Align}}$ encourages the model to maintain the output logits of the retained dataset from the previous forgetting stage, thereby preserving the performance on the retained dataset.

Methods	100-20			80-20				60-20				40-20				Average		
	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$A_o\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$A_o\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$A_o\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$
Pre-train	-	74.6	74.6	-	72.9	70.9	-	-	71.9	69.7	-	-	72.7	71.3	-	-	73.1	71.6
Retrain	18.4	10.5	0.3	16.0	9.1	0.8	**0.0**	16.9	9.6	**0.0**	0.0	23.4	14.0	0.5	**0.0**	18.7	10.8	0.4
LwF	66.2	60.9	2.1	64.6	60.8	2.1	0.5	64.9	61.4	1.4	**0.0**	65.0	60.7	1.4	**0.0**	65.2	61.0	1.8
EWC	66.9	61.0	0.4	66.0	62.9	1.5	**0.0**	66.2	63.9	1.2	**0.0**	64.8	59.7	0.5	**0.0**	66.0	61.9	0.9
MAS	66.6	60.7	0.7	65.4	61.8	1.6	**0.0**	66.1	63.5	0.8	**0.0**	64.2	58.6	0.3	**0.0**	65.6	61.2	0.9
FDR	64.4	59.3	4.1	62.2	58.0	3.9	**0.0**	65.0	62.8	2.4	**0.0**	65.7	62.6	2.3	**0.0**	64.3	60.7	3.3
DER++	66.1	62.8	4.8	63.8	61.7	5.0	**0.0**	64.4	61.6	2.4	**0.0**	65.0	61.8	2.7	**0.0**	64.8	62.0	3.7
LIRF	28.6	60.1	55.8	28.3	58.5	52.2	43.4	35.8	56.1	43.5	33.5	36.8	59.8	44.7	22.1	32.4	58.6	49.1
SCRUB	67.8	63.1	1.3	66.3	64.4	2.6	**0.0**	66.7	64.5	0.8	**0.0**	68.4	66.9	1.5	**0.0**	67.3	64.7	1.6
SCRUB-S	71.2	69.6	1.7	69.1	70.4	3.0	9.0	70.4	71.9	0.8	6.4	70.1	70.1	1.1	2.3	70.2	70.5	1.7
GS-LoRA	71.6	72.1	3.5	68.4	71.1	4.9	**0.0**	69.7	72.0	2.2	**0.0**	70.2	71.0	1.8	**0.0**	70.0	71.6	3.1
Ours	**74.5**	**74.4**	**0.1**	**74.3**	**74.2**	**0.0**	**0.0**	**74.2**	**75.3**	**0.0**	**0.0**	**75.8**	**75.9**	**0.0**	**0.0**	**74.7**	**75.0**	**0.1**

Table 1: Performance comparison on CASIA-Face100 dataset. The best results are highlighted in bold. H is the harmonic mean of A_r and A_f, where A_r is the accuracy on the retained set, A_f is the accuracy on the forgotten set, and A_o is the accuracy on the old forgotten set. The forgetting tasks are denoted as X-Y, where X indicates the total number of classes remmbered by model and Y indicates the number of classes to be forgotten.

The overall loss function for the continual forgetting task is defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{Forget}} + \mathcal{L}_{\text{Retain}} + \lambda \mathcal{L}_{FKIL} + \beta \mathcal{L}_{\text{Align}}, \quad (16)$$

where λ and β are hyperparameters that control the balance between the different loss terms which are emperically set to 0.1 and 1 respectively.

Experiments

Experimental Setup

In this section, we introduce the experimental setup. We conduct experiments on CASIA-Face100 (Zhao et al. 2024) and ImageNet100 (Deng et al. 2009) datasets. More details are provided in the supplementary material.

Evaluation metrics. We use the harmonic mean of accuracy on the retained set and forgetting set as our evaluation metric, denoted as H. Specifically, we define A_r as the accuracy on the retained set, A_f as the accuracy on the forgetting set, and A_o as the accuracy on the old forgetting set. The harmonic mean is calculated as:

$$H = \frac{2 \cdot A_r \cdot (A_{ori} - A_f)}{A_r + (A_{ori} - A_f)}, \quad (17)$$

where A_{ori} denotes the pretrained model's accuracy on the forgetting set. We report accuracy on retained, forgetting, and old forgetting sets for comprehensive evaluation. Higher H indicates better overall performance, while lower A_f and A_o indicate better forgetting effectiveness.

Comparison methods. We compare our method with various state-of-the-art methods, including machine unlearning methods (e.g., DELETE (Zhou et al. 2025), SCRUB (Kurmanji et al. 2023), SCRUB-S (Kurmanji et al. 2023), LIRF (Zhao et al. 2024)), continual learning methods (e.g., L2 (Zhao et al. 2024), EWC (Kirkpatrick et al. 2017), MAS (Aljundi et al. 2018), DER++ (Buzzega et al. 2020), FDR (Zhao et al. 2024), LwF (Feng and Darrell 2015)), and continual forgetting method GS-LoRA (Zhao et al. 2024). All methods use the same pretrained model and the retrain method is trained only on replay data.

Comparison Results

In this section, we present the comparison results of our method with state-of-the-art methods on two public datasets.

Results on CASIA-Face100. As shown in Table 1, our method achieves the best performance on all tasks in the CASIA-Face100 dataset. Particularly, in the 100-20 task, our method reaches an H value of 74.5, significantly outperforming other methods. Benefiting from the forgetting knowledge layer localization module and forgetting knowledge parameter isolation module, our method can effectively forget target knowledge while maintaining good performance on retained knowledge. Additionally, our method achieves low accuracy on the forgetting set, demonstrating its ability to precisely forget target knowledge. We also conduct a single-word forgetting task performance comparison, as shown in Table 3. The results demonstrate that our method can achieve precise forgetting of target knowledge while retaining good performance on retained knowledge.

Results on ImageNet100. The results shown in Table 2 demonstrate that our method achieves the best overall performance on all tasks in the ImageNet100 dataset. Our method achieves the best balance between forgetting and retaining knowledge. Specifically, in the final task, our method achieves an H value of 87.0, which is significantly higher than other methods. This indicates that our method can effectively forget target knowledge while mitigating over-forgetting of retained knowledge.

Ablation Study

We conduct ablation studies on the ImageNet100 dataset to analyze the effectiveness of each component, the impact of

Methods	100-20			80-20				60-20				40-20				**Average**		
	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$A_o\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$A_o\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$A_o\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$
Pre-train	-	90.0	82.9	-	90.1	89.6	-	-	89.7	90.9	-	-	92.8	86.5	-	-	90.6	87.5
Retrain	48.4	34.2	**0.0**	55.3	40.0	**0.0**	**0.0**	58.8	46.6	11.2	**0.0**	68.5	62.2	10.3	1.1	57.8	45.7	5.4
EWC	79.7	76.8	**0.0**	82.6	76.7	0.0	0.7	84.2	78.7	0.3	1.9	83.3	80.4	0.1	2.3	82.5	78.2	0.1
MAS	78.0	73.7	**0.0**	81.4	74.5	**0.0**	0.2	82.8	76.2	0.3	1.9	80.5	75.2	0.0	2.6	80.7	74.9	0.1
LwF	83.8	85.2	0.4	87.0	84.8	0.3	5.1	87.7	85.0	0.4	2.4	86.4	86.4	0.0	1.8	86.2	85.3	0.3
DER++	83.6	**89.4**	4.4	82.7	**90.2**	13.1	62.8	87.3	**89.7**	5.8	28.2	**88.7**	**93.8**	2.3	23.9	85.6	**90.8**	6.5
FDR	28.0	16.8	0.2	37.2	23.5	0.3	0.1	35.5	22.3	2.6	0.9	52.6	38.1	1.4	1.0	38.3	25.2	1.1
SCRUB	72.2	63.9	**0.0**	74.1	63.2	**0.0**	**0.0**	74.9	63.7	**0.0**	**0.0**	76.0	67.7	**0.0**	**0.0**	74.3	64.6	**0.0**
SCRUB-S	79.0	75.5	**0.0**	81.5	74.9	0.2	0.2	77.6	67.7	**0.0**	0.1	72.4	62.3	**0.0**	0.0	77.6	70.1	0.1
DELETE	82.2	88.6	6.2	86.9	84.4	0.1	70.7	85.0	80.1	0.3	67.6	86.5	86.5	**0.0**	73.7	85.2	84.9	1.7
GS-LoRA	84.2	85.6	**0.0**	86.8	84.3	0.1	0.5	87.1	83.6	**0.0**	0.2	85.4	84.3	**0.0**	0.4	85.9	84.4	0.0
Ours	**85.0**	87.5	0.3	**88.3**	87.6	0.6	3.4	**88.6**	86.5	**0.0**	3.0	86.1	85.8	**0.0**	**0.0**	**87.0**	86.8	0.2

Table 2: Continual forgetting performance comparison on ImageNet100 dataset. The best results are highlighted in bold.

Methods	100-5			100-10			100-50			100-90			**Average**		
	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$
Pre-train	-	73.9	70.9	73.8	72.7	-	72.5	74.9	-	72.3	73.9	-	73.1	73.1	-
Retrain	16.5	9.3	**0.0**	18.0	10.3	**0.0**	19.7	11.4	**0.0**	46.5	33.9	0.0	25.2	16.2	0.0
LwF(ICCV'15)	68.0	68.6	**0.0**	70.1	67.6	**0.0**	63.8	55.6	0.0	61.3	52.3	0.0	65.8	61.0	**0.0**
EWC(PNAS'17)	69.7	68.7	0.2	69.2	65.9	**0.0**	62.5	53.6	0.0	44.4	31.8	0.0	61.4	55.0	0.1
MAS(ECCV'18)	69.7	69.3	0.7	69.4	66.4	0.1	62.7	53.9	0.0	46.8	34.2	0.0	62.1	56.0	0.2
FDR(ICLR'19)	70.3	69.7	**0.0**	70.0	67.5	**0.0**	65.4	58.0	0.0	53.9	42.4	0.0	64.9	59.4	**0.0**
DER++(NIPS'20)	69.7	68.6	**0.0**	70.6	68.6	**0.0**	64.6	56.8	0.0	61.4	52.5	0.0	66.6	61.6	**0.0**
LIRF(ECCV'22)	25.6	67.7	55.1	26.4	65.8	56.3	47.1	59.0	35.6	54.5	44.3	3.1	38.4	59.2	37.5
SCRUB(NIPS'23)	67.8	64.9	**0.0**	68.3	64.3	**0.0**	65.8	58.7	0.0	16.1	9.0	0.0	54.5	49.3	**0.0**
SCRUB-S(NIPS'23)	70.3	70.0	0.2	71.2	69.8	0.1	57.5	51.2	9.3	17.5	9.9	0.0	54.1	50.2	2.4
GS-LoRA(CVPR'24)	71.0	71.2	**0.0**	71.8	70.8	**0.0**	71.3	68.1	0.0	73.7	73.6	0.0	71.9	70.9	**0.0**
Ours	**72.7**	**74.5**	**0.0**	**73.2**	**73.7**	**0.0**	**73.2**	**70.7**	**0.0**	**74.0**	**74.2**	**0.0**	**73.3**	**73.3**	**0.0**

Table 3: Singele-task forgetting performance comparison on CASIA-Face100 dataset. The best results are highlighted in bold.

#	FKLL	FKPI	$\mathcal{L}_{FKIL}$	$\mathcal{L}_{\text{Align}}$	$H\uparrow$	$Acc_r\uparrow$	$Acc_f\downarrow$
0					84.79	83.25	**0.00**
1	✓				83.06	82.21	0.08
2		✓			86.16	86.00	0.13
3	✓	✓			85.23	84.10	0.22
4	✓	✓	✓		85.17	83.29	0.13
5	✓	✓		✓	86.19	85.69	0.68
6	✓	✓	✓	✓	**87.01**	**86.84**	0.23

Table 4: Ablation study on the components of our method.

	$H\uparrow$	$A_r\uparrow$	$A_f\downarrow$	$A_o\downarrow$
Initialization Methods				
Zero	82.98	79.97	1.15	7.21
LoRA	72.98	63.54	0.08	0.87
Selection Methods				
Random	81.37	76.13	**0.03**	7.18
Fix	85.01	83.47	0.80	7.57
Ours	**87.01**	**86.84**	0.23	**2.13**

Table 5: Ablation study on the impact of initialization methods and selection methods.

LoRA ranks and selected layers, and the effectiveness of LoRA initialization and layer localization strategies.

Effectiveness of each component. To analyze the effectiveness of each component in our method, we conduct ablation studies on the ImageNet100 dataset. The results are shown in Table 4. We observe that each component contributes to the overall performance of our method. Specifically, the FKLL and FKPI significantly improve forgetting performance while maintaining good performance on retained knowledge. The $\mathcal{L}_{FKIL}$ further enhances the forgetting performance by pushing forgetting features away from retained features. Overall, our method achieves a good balance between forgetting and retaining knowledge.

Numbers of LoRA ranks, mixture factor and selected layers. To analyze the impact of the number of LoRA ranks and selected layers, we conduct experiments with different ranks and layer selections. The results are shown in

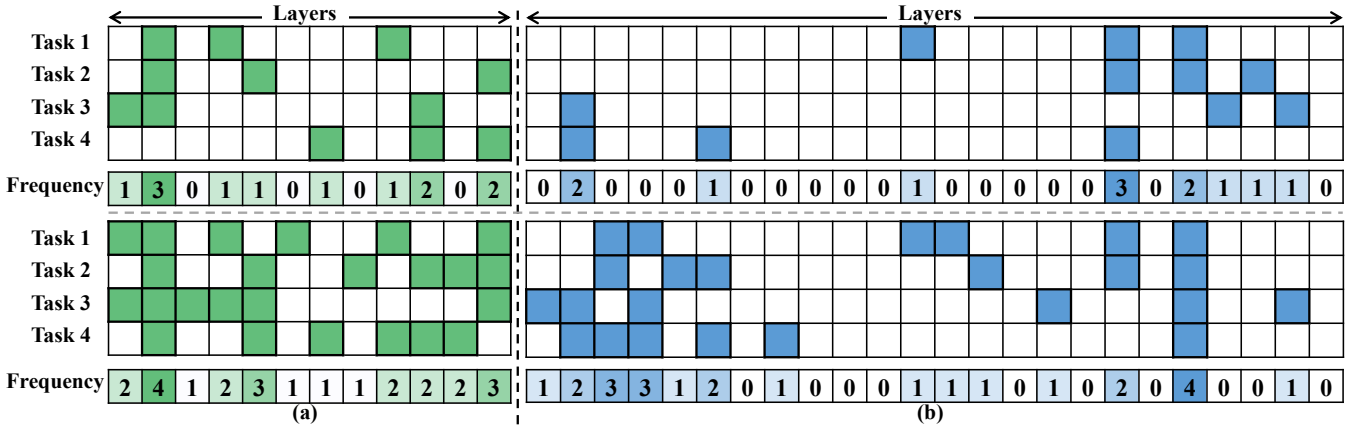

Figure 3: The distribution of selected layers across tasks in the pre-trained model.The red line indicates the frequency distribution selected by our method. The green blocks indicate the selected layer in each face recognition task (a). The blue blocks indicate the selected layer in each image classification task (b).

Figure 4. We observe that increasing the number of LoRA ranks generally leads to better forgetting performance, as it allows for more flexible adaptation to the forgetting knowledge. However, too many ranks may also introduce noise, leading to performance degradation. Figure 4(b) showed as $alpha$ increases, FKPI focuses more on identifying forgetting knowledge. To balance the forgetting knowledge and retained knowledge, we set it to 0.4.Similarly, selecting an appropriate number of layers is crucial for achieving a balance between forgetting and retaining knowledge. To balance the forgetting performance and the computational cost, we set the LoRA rank to 8 and select 12 layers in our experiments.

Effectiveness of LoRA initialization. To analyze the effectiveness of LoRA initialization, we conduct experiments with different initialization methods. The results are shown in Table 5. We observe that initializing LoRA with the pre-trained weights significantly improves forgetting performance compared to random initialization. This indicates that leveraging pre-trained knowledge to promote adaptation of forgetting tasks while preserving retained knowledge.

Effectiveness of layer localization. To analyze the effectiveness of layer localization, we show the sensitivity distribution of forgetting knowledge across layers in Figure 3. We observe that the sensitivity of forgetting knowledge varies significantly across layers, with some layers being more sensitive to forgetting knowledge than others. By selecting layers that are more sensitive to forgetting knowledge, our method can achieve better forgetting performance while minimizing the impact on remaining knowledge. Based on these observations, we select the top 12 layers with the highest forgetting knowledge sensitivity as fixed selection mehtod to campare with our adaptively localization strategy in Table 5.

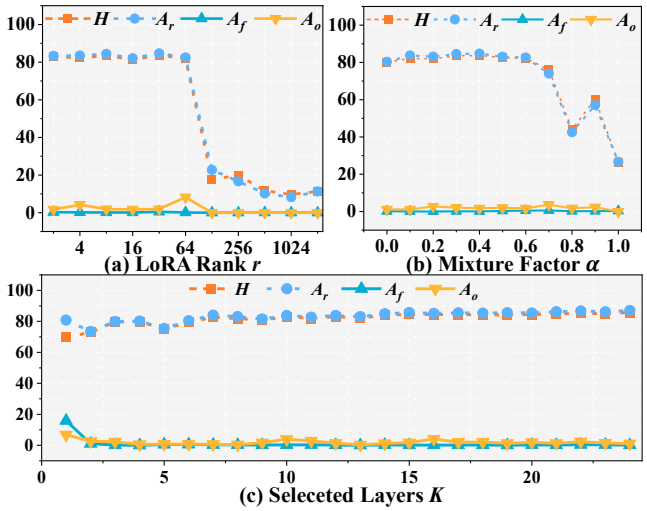

Figure 4: Ablation study on the number of LoRA ranks (a), mixture factor (b) and selected layers (c).

Conclusion

In this paper, we propose the forgetting knowledge localization and isolation method for continual forgetting of pre-trained vision models, addressing the challenges of efficiently forgetting specific knowledge while preserving remaining knowledge integrity. Our method introduces forgetting knowledge localization module to achieve precisely forgetting by selecting layers related to forgetting knowledge and forgetting knowledge parameter isolation module to isolate parameters that can be fine-tuned to remove forgetting knowledge. The forgetting knowledge isolation loss and retained knowledge alignment loss are introduced to further mitigate feature drift and over-forgetting of remaing knowledge. Extensive experiments demonstrate that our method achieves state-of-the-art performance on both forgetting and retained datasets across multiple continual forgetting tasks. Future work will explore applications of our method in larer vision foundation models.

Acknowledgments

This work was supported by the National Nature Science Foundation of China (62322211), the "Pioneer" and "Leading Goose" R&D Program of Zhejiang Province (2024C01023, 2024C01107, 2023C01030, 2023C01046) and the National Key Research and Development Program of China under Grant (2023YFB4502800).

References

Aljundi, R.; Babiloni, F.; Elhoseiny, M.; Rohrbach, M.; and Tuytelaars, T. 2018. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*, 139–154.

Baumhauer, T.; Schöttle, P.; and Zeppelzauer, M. 2022. Machine unlearning: Linear filtration for logit-based classifiers. *Machine Learning*, 111(9): 3203–3226.

Bourtoule, L.; Chandrasekaran, V.; Choquette-Choo, C. A.; Jia, H.; Travers, A.; Zhang, B.; Lie, D.; and Papernot, N. 2021. Machine unlearning. In *2021 IEEE symposium on security and privacy (SP)*, 141–159. IEEE.

Brophy, J.; and Lowd, D. 2021. Machine unlearning for random forests. In *International Conference on Machine Learning*, 1092–1104. PMLR.

Buzzega, P.; Boschini, M.; Porrello, A.; Abati, D.; and Calderara, S. 2020. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33: 15920–15930.

Cao, Y.; Peng, H.; Yu, Z.; and Yu, P. S. 2024. Hierarchical and incremental structural entropy minimization for unsupervised social event detection. In *Proceedings of the AAAI conference on artificial intelligence*, 8255–8264.

Chen, Y.; Xiong, J.; Xu, W.; and Zuo, J. 2019. A novel online incremental and decremental learning algorithm based on variable support vector machine. *Cluster Computing*, 22: 7435–7445.

Crawford, K.; and Paglen, T. 2021. Excavating AI: The politics of images in machine learning training sets. *Ai & Society*, 36(4): 1105–1116.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Feng, J.; and Darrell, T. 2015. Learning The Structure of Deep Convolutional Networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Gao, C.; Wang, L.; Ding, K.; Weng, C.; Wang, X.; and Zhu, Q. 2024. On large language model continual unlearning. *arXiv preprint arXiv:2407.10223*.

GDPR, E. 2018. General data protection regulation (gdpr).

Golatkar, A.; Achille, A.; and Soatto, S. 2020. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9304–9312.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W.; et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2): 3.

Huu-Tien, D.; Thanh-Tung, H.; Nguyen, L.-M.; and Inoue, N. 2025. Improving the Robustness of Representation Misdirection for Large Language Model Unlearning. *arXiv preprint arXiv:2501.19202*.

Izzo, Z.; Smart, M. A.; Chaudhuri, K.; and Zou, J. 2021. Approximate data deletion from machine learning models. In *International Conference on Artificial Intelligence and Statistics*, 2008–2016. PMLR.

Jia, J.; Zhang, Y.; Zhang, Y.; Liu, J.; Runwal, B.; Diffenderfer, J.; Kailkhura, B.; and Liu, S. 2024. Soul: Unlocking the power of second-order optimization for llm unlearning. *arXiv preprint arXiv:2404.18239*.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13): 3521–3526.

Kurmanji, M.; Triantafillou, P.; Hayes, J.; and Triantafillou, E. 2023. Towards unbounded machine unlearning. *Advances in neural information processing systems*, 36: 1957–1987.

Li, L.; Cong, G.; Qi, Y.; Zha, Z.-J.; Wu, Q.; Sheng, Q. Z.; Huang, Q.; and Yang, M.-H. 2025. Dubbing Movies via Hierarchical Phoneme Modeling and Acoustic Diffusion Denoising. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Li, L.; Gao, X.; Deng, J.; Tu, Y.; Zha, Z.-J.; and Huang, Q. 2022. Long short-term relation transformer with global gating for video captioning. *IEEE Transactions on Image Processing*, 31: 2726–2738.

Li, M.; Xu, X.; Fan, H.; Zhou, P.; Liu, J.; Liu, J.-W.; Li, J.; Keppo, J.; Shou, M. Z.; and Yan, S. 2023. STPrivacy: Spatio-temporal privacy-preserving action recognition. In *ICCV*.

Lin, J.; Wu, Z.; Lin, W.; Huang, J.; and Luo, R. 2024. M2sd: Multiple mixing self-distillation for few-shot class-incremental learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3422–3431.

Liu, J.; Ke, W.; Wang, P.; Shang, Z.; Gao, J.; Li, G.; Ji, K.; and Liu, Y. 2024a. Towards continual knowledge graph embedding via incremental distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 8759–8768.

Liu, X.; Li, L.; Wang, S.; Zha, Z.-J.; Li, Z.; Tian, Q.; and Huang, Q. 2022. Entity-enhanced adaptive reconstruction network for weakly supervised referring expression grounding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3): 3003–3018.

Liu, Y.; An, J.; Zhang, W.; Li, M.; Wu, D.; Gu, J.; Lin, Z.; and Wang, W. 2024b. RealEra: Semantic-level Concept Erasure via Neighbor-Concept Mining. *arXiv preprint arXiv:2410.09140*.

Lu, X.; Yuan, Z.; Zhang, Y.; Ai, H.; Cheng, S.; Ge, Y.; Fang, F.; and Chen, N. 2024. A comparison of statistical learning of naturalistic textures between DCNNs and the human visual hierarchy. *Science China Technological Sciences*, 67(8): 2310–2318.

Mahadevan, A.; and Mathioudakis, M. 2021. Certifiable machine unlearning for linear models. *arXiv preprint arXiv:2106.15093*.

Paglen, T. 2020. Imagenet roulette–trevor paglen.

Sekhari, A.; Acharya, J.; Kamath, G.; and Suresh, A. T. 2021. Remember what you want to forget: Algorithms for machine unlearning. *Advances in Neural Information Processing Systems*, 34: 18075–18086.

Shan, L.; Zhou, W.; Li, W.; and Ding, X. 2024. Lifelong Learning and Selective Forgetting via Contrastive Strategy. *arXiv preprint arXiv:2405.18663*.

Shibata, T.; Irie, G.; Ikami, D.; and Mitsuzumi, Y. 2021. Learning with Selective Forgetting. In *IJCAI*, volume 2, 6.

Sun, N.; Wang, N.; Wang, Z.; Nie, J.; Wei, Z.; Liu, P.; Wang, X.; and Qu, H. 2023. Lazy machine unlearning strategy for random forests. In *International Conference on Web Information Systems and Applications*, 383–390. Springer.

Thakral, K.; Glaser, T.; Hassner, T.; Vatsa, M.; and Singh, R. 2025. Continual unlearning for foundational text-to-image models without generalization erosion. *arXiv preprint arXiv:2503.13769*.

Tu, Y.; Li, L.; Su, L.; Zha, Z.-J.; and Huang, Q. 2024. Smart: Syntax-calibrated multi-aspect relation transformer for change captioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(7): 4926–4943.

Wang, G.; Bao, H.; Liu, Q.; Zhou, T.; Wu, S.; Huang, T.; Yu, Z.; Lu, C.; Gong, Y.; Zhang, Z.; et al. 2024. Brain-inspired artificial intelligence research: A review. *Science China Technological Sciences*, 67(8): 2282–2296.

Xu, J.; Wu, Z.; Wang, C.; and Jia, X. 2024. Machine unlearning: Solutions and challenges. *IEEE Transactions on Emerging Topics in Computational Intelligence*.

Zhai, J.-T.; Liu, X.; Yu, L.; and Cheng, M.-M. 2024. Fine-grained knowledge selection and restoration for non-exemplar class incremental learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 6971–6978.

Zhang, B.; Li, L.; Wang, S.; Cai, S.; Zha, Z.-J.; Tian, Q.; and Huang, Q. 2024a. Inductive state-relabeling adversarial active learning with heuristic clique rescaling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhang, X.; Chen, Y.; Ma, C.; Fang, Y.; and King, I. 2024b. Influential exemplar replay for incremental learning in recommender systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 9368–9376.

Zhao, H.; Ni, B.; Fan, J.; Wang, Y.; Chen, Y.; Meng, G.; and Zhang, Z. 2024. Continual forgetting for pre-trained vision models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 28631–28642.

Zheng, J.; Liu, X.; Liu, W.; He, L.; Yan, C.; and Mei, T. 2022. Gait Recognition in the Wild with Dense 3D Representations and A Benchmark. In *CVPR*, 20228–20237.

Zheng, J.; Liu, X.; Wang, S.; Wang, L.; Yan, C.; and Liu, W. 2023. Parsing is All You Need for Accurate Gait Recognition in the Wild. In *ACMMM*, 116–124.

Zhou, Y.; Zheng, D.; Mo, Q.; Lu, R.; Lin, K.-Y.; and Zheng, W.-S. 2025. Decoupled distillation to erase: A general unlearning method for any class-centric tasks. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 20350–20359.