RAEncoder: A Label-Free Reversible Adversarial Examples Encoder for Dataset Intellectual Property Protection

Fan Xing, Zhuo Tian, Xuefeng Fan, Xiaoyi Zhou*
Hainan University
{xingfan, tianzhuo, xffan, xy.zhou}@hainanu.edu.cn

Abstract

Reversible Adversarial Examples (RAE) are designed to protect the intellectual property of datasets. Such examples can function as imperceptible adversarial examples to erode the model performance of unauthorized users while allowing authorized users to remove the adversarial perturbations and recover the original samples for normal model training. With the rise of Self-Supervised Learning (SSL), an increasing number of unlabeled datasets and pre-trained encoders are available in the community. However, existing RAE methods not only rely on well-labeled datasets for training Supervised Learning (SL) models but also exhibit poor adversarial transferability when attacking SSL pre-trained encoders. To address these challenges, we propose RAEncoder, the first framework for RAEs without the need for labeled samples. RAEncoder aims to generate universal adversarial perturbations by targeting SSL pre-trained encoders. Unlike traditional RAE approaches, the pre-trained encoder outputs the feature distribution of the protected dataset rather than classification labels, enhancing both the attack success rate and transferability of RAEs. Extensive experiments are conducted on six pre-trained encoders and four SL models, covering aspects such as imperceptibility and transferability. Our results demonstrate that RAEncoder effectively protects unlabeled datasets from malicious infringements. Additional robustness experiments further confirm the security of RAEncoder in practical application scenarios.

1. Introduction

Reversible Adversarial Examples (RAE) [20, 31–34, 37, 38, 41] are highlighted as a fundamental method for protecting dataset Intellectual Property (IP). Unlike traditional methods for dataset IP protection, such as reversible watermarking [10, 19, 42] and image encryption techniques[11, 35], RAE transforms the dataset into adversarial examples that can mislead model training while allowing authorized users to remove the adversarial perturbations and restore the original dataset [20]. The adversarial attack methods used in existing RAE schemes focus on targeting specific Supervised Learning (SL) classifiers through optimization-based or gradient-based attacks, which result in a notable limitation: they lose adversarial strength when attacking different model architectures, and the adversarial strength diminishes especially when attacking downstream models trained with Self-Supervised Learning (SSL) pre-trained encoders.

SSL is an emerging paradigm in machine learning. It typically involves pre-training a generic encoder on vast amounts of unlabeled data, and it can then serve as a feature extractor for a wide range of downstream tasks. By utilizing a pre-trained encoder, any resource-constrained user can leverage the advantages of a large model without the need to undertake expensive training from scratch. Considering SSL for RAEs is highly noteworthy because if RAEs exhibit poor transferability when attacking SSL pre-trained encoders, any attacker can leverage them to pre-process RAEs, thereby mitigating the impact of them on the training performance of unauthorized downstream models.

Due to the fact that SSL pre-trained encoders learn feature representations of the data directly rather than feature-label relationships, adversarial attacks against SSL pre-encoders can exhibit excellent transferability across different downstream model architectures [43, 44]. Although this attack process demonstrates significant effectiveness, using it as a method to generate RAEs for protecting dataset IP introduces an additional challenge. Specifically, the protection scheme must address the worst-case scenario—adaptive attacks, where the attacker is assumed to be aware of the detailed steps of the protection scheme [32]. This requires the protection scheme to effectively differentiate between authorized and unauthorized access. To differentiate between authorized and unauthorized users, existing Reversible Data Hidding (RDH)-based RAE methods embed encrypted information in RAEs, ensuring that only users with the correct key can recover original samples. However, embedding any information within RAEs could significantly impact both adversarial strength and vi-

sual quality[20, 31, 32]. Then, without embedding information, how can RAEs be designed to differentiate user authorization and ensure both effectiveness and robustness when attacking SSL encoders?

To bridge the gap between the practical applicability of RAE and real-world needs in the context of SSL's growing popularity, we propose a novel approach, RAEncoder. To be specific, we design a noise encoder with a fixed signature noise as input. It learns the data distribution and generates universal adversarial perturbations by targeting SSL encoder. To effectively differentiate between authorized and unauthorized users, an interference pattern is designed to make the noise encoder insensitive to inputs other than the correct signature noise. By inputting the correct signature noise, authorized users can employ the noise encoder to obtain the recovered perturbation, thereby recovering the original sample. During noise encoder training, we employ high-frequency filters [21, 44] to optimize RAE imperceptibility, as high-frequency perturbations are less detectable than low-frequency ones.

Compared to the existing work, our contributions are as follows:

- **A Novel RAEncoder Framework**: We introduce RAEncoder to address the limitations of existing RAE frameworks in the context of the growing popularity of SSL. To the best of our knowledge, RAEncoder is the first RAE framework applicable for protecting the IP of unlabeled datasets with 100% fidelity.
- **Innovative Noise Encoder**: A noise encoder capable of generating RAEs guided by sample texture features is devised to successfully balance imperceptibility with adversarial strength. Additionally, we introduce three losses in the interference pattern to guide adversarial attacks on SSL pre-encoders, ensuring the RAE's robustness.
- **Extensive Experimental Validation**: Through extensive experiments, we compare our RAEncoder with state-of-the-art (SOTA) RAE schemes. The results demonstrate that our RAEncoder outperforms existing methods in various metrics, including imperceptibility, adversarial strength, and transferability.

2. Related Work

2.1. SSL and Its Emerging Threats

SSL uses the inherent information present in unlabeled data to pre-train encoders. These encoders learn to transform complex inputs into generic representations. Once trained, they can extract valuable domain knowledge and serve as versatile tools for solving different specific tasks. With the thriving development of SSL, pre-trained encoders have gained popularity in the industry. Many service providers openly release their pre-trained encoders, such as Google's BYOL [12] and Meta's MoCo [6, 14]. Recently, an increasing number of studies begin to investigate adversarial attacks against pre-trained encoders in SSL. Zhou *et al*. proposed the first attack framework that generates downstream-agnostic adversarial examples based on a cross-modal pre-trained encoder for images and text [43]. Zhou *et al*. proposed the first scheme for generating universal adversarial examples based on pre-trained encoders[44].

2.2. Reversible Adversarial Examples

Existing RAE schemes can be divided into two categories, RDH-based methods[20, 31–34, 37] and generative methods [38, 41], as shown in Figure 1. To generate adversarial examples, recent research endeavors utilize a range of adversarial attacks targeting SL classifier, such as I-FGSM[18], DeepFool[23], and C&W[1].

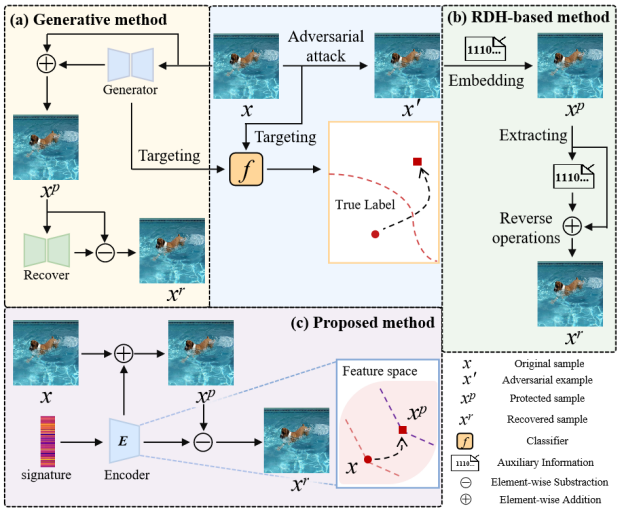


Figure 1. Existing methods can be divided into two categories: generative and RDH-based. Our method not only eliminates reliance on classifiers for adversarial attacks but also achieves 100% fidelity without compromising adversarial strength.

Given that images contain inherent structures that allow adversarial attacks to be reversed [22], the concept of RAE was first introduced [20] and proposed the first RAE generation method based on RDH technology. Their approach involves embedding the pixel differences between adversarial examples and the original image into the image using RDH, allowing authorized users to achieve 100% recovery of original samples by extracting the pixel differences. However, this method still has several issues, such as the deterioration of the visual quality of adversarial samples and a reduction in adversarial strength due to embedding information. Thus, reducing the embedded perturbation amount in RAE is a feasible approach. Yin *et al*. confined perturbations to regions most impactful on adversarial strength[34]. They leveraged the characteristic of the YUV color space, where the human eye is less sensitive to changes in the UV

channels, by embedding the perturbation differences of the Y channel into the UV channels. Zhang *et al*. leveraged beam search-based black-box attacks in combination with reversible data hiding featuring grayscale invariance[37]. Xiong *et al*. proposed PGN, which reduces adversarial perturbations to a sign matrix format and a product with fixed perturbation parameters[31]. Therefore, PGN only needs to record the sign matrix corresponding to pixel differences and embed it with the fixed perturbation parameters, significantly reducing the amount of embedding required. Meanwhile, existing work combined Reversible Image Transformation (RIT)[40] with RDH which involves embedding the reversible transformation information between the original sample and the adversarial example, rather than pixel differences, into the RAEs[32, 33].

In the field of generative methods, SRAE [38] developed RAEs using a recoverable generative adversarial network designed to minimize recovery errors while maximizing their effectiveness in adversarial attacks. GEAA [41] utilizes a dual-stream architecture to generate adversarial and recovered samples, ensuring that the distribution of adversarial samples significantly differs from the original.

In summary, RDH-based methods preserve the inverse operation that restores adversarial examples to the original samples, allowing most RDH-based methods to achieve 100% sample recovery. While generative methods cannot currently achieve 100% recovery, they do not require any information embedding that might affect adversarial strength. As a result, generative methods exhibit significantly higher adversarial strength compared to RDH-based methods.

It is noteworthy that existing work generates RAEs by targeting supervised learning models for adversarial attacks, without addressing the new challenges posed by the popularity of SSL. Consequently, existing RAE solutions are unable to effectively protect unlabeled datasets.

3. Methodology

3.1. Problem Definition

Given a pre-trained encoder $e_\theta(\cdot)$ with input x_{pt} in $\mathcal{D}_{pt}$, where θ represents the parameters of the pre-trained encoder. $\mathcal{D}_{pt}$ denotes the pre-training dataset. The data owner uses the original sample x in original dataset $\mathcal{D}$ (remains unattainable for the attacker) to generate universal adversarial perturbation δ against the pre-trained encoder. The universal adversarial perturbation δ should be sufficiently small and modeled by an upper-bound ϵ on the l_p norm. The task of finding a perturbation can be formulated as:

$$e_\theta(x+\delta) \neq e_\theta(x), \qquad s.t. ||\delta||_p \leq \varepsilon \tag{1}$$

where $||\cdot||_p$ is the l_p norm.

The data owner's objective is to achieve a universal non-targeted attack to deceive the downstream classifier $f_{\theta'}$. Therefore, the data owner's objective can be formalized as:

$$f_{\theta'}(e_\theta(x+\delta)) \neq f_{\theta'}(e_\theta(x)), \qquad s.t. ||\delta||_p \leq \varepsilon \tag{2}$$

When all original samples x in original dataset $\mathcal{D}$ are augmented with a universal adversarial perturbation δ, it results in a protected dataset $\mathcal{D}_p$ composed of protected samples (i.e., reversible adversarial examples) x^p. The $\mathcal{D}_p$ is used as a reference for all users. We assume that any users have access to public pre-trained encoders $e_\theta^{pub}(\cdot)$ with input x^{pub} in the public dataset $\mathcal{D}^{pub}$, which they can obtain through purchase or direct download from public repositories.

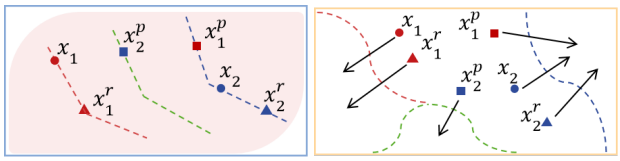
The attackers' objective is to encode protected datasets $\mathcal{D}_p$ using $e_\theta^{pub}(\cdot)$ to minimize the impact of $\mathcal{D}_p$ on training performance and then training downstream classifier $f_\theta^{atk}(e_\theta^{pub}(\cdot))$. However, as shown in Figure 2, $x^p \in \mathcal{D}_p$ is incorrectly encoded into other feature distributions, resulting in a significant degradation in the performance of the obtained downstream classifier which can be formalized as:

$$\mathcal{A}(f_\theta^{atk}(e_\theta^{pub}(x^p))) \ll \mathcal{A}(f_\theta^{atk}(e_\theta^{pub}(x))) \tag{3}$$

where $\mathcal{A}(\cdot)$ denotes the inference accuracy for a classifier.

For authorized users, they can obtain recovered samples $x^r \in \mathcal{D}_r$ which closely resemble the feature distribution of x for normal downstream task training that can be formalized as:

$$\mathcal{A}(f_\theta^{atk}(e_\theta^{pub}(x^r))) \approx \mathcal{A}(f_\theta^{atk}(e_\theta^{pub}(x))) \tag{4}$$



(a) (b)

Figure 2. Illustrations of sample attributes: (a) The representation of different samples in the feature space. The dashed lines in different colors represent feature representations of various distributions (b) The fit of different samples within the classification decision boundary. The curved lines in different colors indicate the decision boundaries for different classes.

3.2. Training Pipeline

The training pipeline of RAEncoder is illustrated in Figure 3. It consists of a noise encoder E, a high-frequency filter $\mathcal{H}$, and a target encoder e_θ. Specifically, we generate a universal adversarial perturbation δ applicable to all x in D by inputting a fixed signature noise $\mathcal{S}$ into noise encoder E. We then paste δ to x to obtain protected sample x^p in D_p, where high-frequency information of x^p is reduced by high-frequency filter $\mathcal{H}$. Authorized users can input the signature noise $\mathcal{S}$ into the noise encoder E to obtain δ. Subtracting δ

from x^p results in a fully recovered sample x^r identical to the original x, which is achieved for the first time in the previous RAE schemes. During the training process of noise encoder E, we propose an interference pattern to incentivize E to distinguish signature noise $\mathcal{S}$ from other noise inputs, aiming to prevent potential attackers from inputting random noise to steal δ for image restoration.

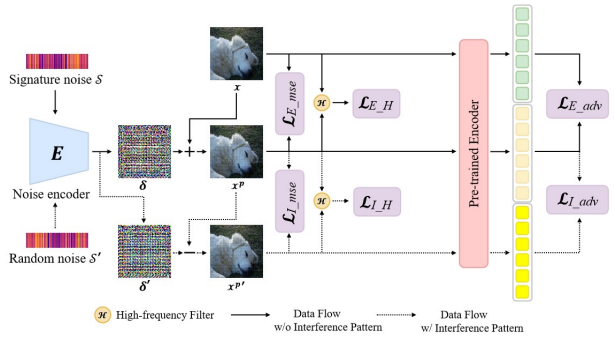


Figure 3. The training pipeline of RAEncoder.

3.2.1. Noise Encoder.

The noise encoder E takes a fixed input $\mathcal{S}$, designated for different authorized users, to generate a universal adversarial perturbation δ, given by:

$$E(\mathcal{S}) = \delta. \tag{5}$$

Applying δ to original samples x yields x^p. The difference between x^p and δ represents the original sample x which can be described as:

$$\delta + x = x^p \tag{6}$$

When authorized users want to recover samples, they only need to input their exclusive $\mathcal{S}$ into E to obtain δ, and original samples can be restored by subtracting δ which can be described as:

$$x^r = x^p - \delta = x \tag{7}$$

The loss function for optimizing noise encoder E can be described as:

$$\mathcal{L}_E = \alpha\mathcal{L}_{E\_adv} + \beta\mathcal{L}_{E\_\mathcal{H}} + \lambda\mathcal{L}_{E\_mse}, \tag{8}$$

where

$$\mathcal{L}_{E\_adv} = \log \frac{\exp(\langle e_\theta(x_i^p), e_\theta(x_i)\rangle/\tau)}{\sum_{k=0}^{N} \exp(\langle e_\theta(x_i^p), e_\theta(x_k)\rangle/\tau)}, \tag{9}$$

$$\mathcal{L}_{E\_\mathcal{H}} = -||\mathcal{H}(x^p) - \mathcal{H}(x)||_2^2, \tag{10}$$

$$\mathcal{L}_{E\_mse} = ||x^p - x||_2^2, \tag{11}$$

$||\cdot||_2$ is the l_2 norm, $\langle u \cdot c\rangle = u^\mathsf{T}c/\|u\|\|c\|$ represents the cosine similarity. i represents the i-th sample, with a maximum value of N, while k represents a number not equal to i. τ indicates a temperature parameter. $\mathcal{L}_{E\_adv}$ drives noise encoder to perform negative pairing between the protected sample x^p and x, maximizing the mutual information distance between their features in the feature space by calculating the InfoNCE loss[4, 24]. $\mathcal{L}_{E\_\mathcal{H}}$ aims to constrain the perturbation intensity in high frequency through the high-frequency filter $\mathcal{H}$. $\mathcal{L}_{E\_mse}$ aims to make sure the indistinguishability between x^p and x.

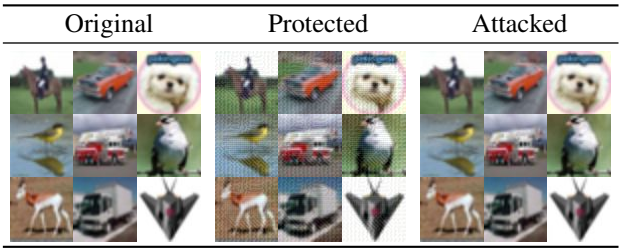


Figure 4. Visual results of original samples different samples generated from noise encoder without interference pattern employed.

3.2.2. Interference Pattern.

Training and optimizing E enables the generation of a highly adversarial universal perturbation δ, significantly affecting the model training effectiveness for unauthorized users. However, we discover that guiding the noise encoder E to solely learn the generation of a universal adversarial perturbation δ from signature noise $\mathcal{S}$ without constraints can lead to vulnerabilities. In the event that the noise encoder E is compromised by attackers, they can potentially acquire a perturbation similar to δ for image recovery by feeding noise of the same size as $\mathcal{S}$ into the noise encoder E. While the likelihood of the noise encoder E being compromised, particularly when stored in cloud services such as DMCP[32], is minimal in practical scenarios, such an outcome remains unacceptable in cases where the noise encoder E cannot distinguish between inputs from authorized users and attackers. As shown in Figure 4, the images shown in the third column are obtained by subtracting the output of noise encoder E when the protected sample x^p and random noise of the same size as signature noise $\mathcal{S}$ are inputted. It can be observed that these samples are very similar to the original images, which completely undermines the security of RAEs. Therefore, an interference pattern is exploited to resolve this problem. The idea is simple: when applying random noise $\mathcal{S}' \neq \mathcal{S}$, noise encoder E should not generate perturbation δ, but instead generate an almost blank perturbation δ' with hardly any pixel values. The subtraction of perturbation δ' from x^p yields $x^{p\prime}$ which can be described as:

$$E(\mathcal{S}') = \delta', \tag{12}$$

$$x^p - \delta' = x^{p'} \tag{13}$$

Thus, the loss function for optimizing noise encoder E with interference pattern can be described as:

$$\mathcal{L}_{E\mathcal{I}} = n\mathcal{L}_{I\_adv} + m\mathcal{L}_{I\_\mathcal{H}} + \gamma\mathcal{L}_{I\_mse} + \mathcal{L}_E, \tag{14}$$

$$\mathcal{L}_{I\_adv} = \log \frac{\exp(\langle e_\theta(x_i^{p'}), e_\theta(x_i)\rangle/\tau)}{\sum_{k=0}^{N} \exp(\langle e_\theta(x_i^{p'}), e_\theta(x_k)\rangle/\tau)}, \tag{15}$$

$$\mathcal{L}_{I\_\mathcal{H}} = -||\mathcal{H}(x^{p'}) - \mathcal{H}(x^p)||_2^2, \tag{16}$$

$$\mathcal{L}_{I\_mse} = ||x^p - x^{p'}||_2^2. \tag{17}$$

$\mathcal{L}_{I\_adv}, \mathcal{L}_{I\_\mathcal{H}}$ and $\mathcal{L}_{I\_mse}$ are employed to minimize δ' generated from random noise $\mathcal{S}'$ towards zero as much as possible.

4. Experiments

4.1. Experimental Setting

4.1.1. Datasets and Models.

We utilize pre-trained encoders from the *solo-learn* [8], a well-established self-supervised learning library, as target encoders. These pre-trained encoders were trained on CIFAR10 [17] or ImageNet [25] datasets, employing ResNet18 [13] as the backbone. Among the array of SSL techniques available, we select six pioneering methods: Barlow Twins [36], BYOL [12], DeepCluster v2 [2], DINO [3], MoCo v2+ [5], and NNCLR [9]. We train VGG16[26], ResNet18, Resnet50[13], and DenseNet121[15] as target models. The datasets STL10 [7], GTSRB [28], CIFAR10, and ImageNet were considered as the protected datasets.

4.1.2. Training configurations.

The noise encoder consists of multiple consecutive deconvolution layers followed by a convolutional layer. The hyperparameters settings for optimizing the noise encoder E are as follows: $n = \alpha = 5$, $m = \beta = 1$, $\gamma = \lambda = 1$ and $\varepsilon = 10/255$, training epochs set to 20 and Adam optimizer [16] is adopted with an initial learning rate of 0.0002. For RDH-based method [32], PGN, and generative method GEAA, SRAE, all experiments conducted were based on their default settings as reported in their respective studies. Since they are not designed for adversarial attacks on pre-trained encoders, for a fair comparison to highlight the advantages of RAEncoder, we connect their target model's classification head to an encoder trained on the same pre-training dataset. Thus, the target model serves as a downstream model connected to the pre-trained encoder. Our experiments are conducted on two NVIDIA GeForce RTX 4090 GPUs.

4.1.3. Evaluation Metrics.

We employ Structural Similarity Index (SSIM) [30] and Peak Signal-to-Noise Ratio (PSNR) to assess the quality of image. We use Attack Success Rate (ASR) to measure the success rate of samples that deceive the classifier. A higher numerical value indicates stronger attack capability.

4.2. Attack Performance

We employ six types of SSL pre-trained encoders as target encoders e_θ. Using two pre-training datasets, CIFAR10 and ImageNet, we generate x^p for four downstream datasets and evaluated ASR of x^p on the corresponding target encoders. It is worth noting that such experimental setups are deployed to investigate whether the RAEs generated by the RAEncoder possess strong adversarial strength to successfully deceive target encoders. As shown in Table 1, across the settings of six pre-trained encoders and four downstream datasets, the ASR of protected samples remains consistently high, with over 80% achieved in almost all settings. Meanwhile, it is evident that protected samples generated by pre-trained encoders trained on corresponding original samples exhibit stronger adversarial attack strength. For example, When the pre-trained dataset is ImageNet and the downstream dataset is also ImageNet, the protected dataset generated by RAEncoder achieved an ASR close to 100% when attacking encoders pre-trained on ImageNet.

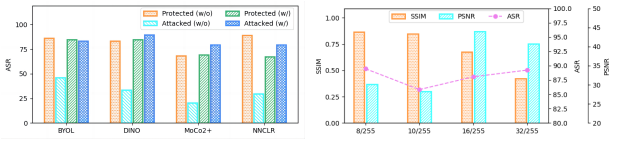
	Balow	BYOL	DeepC2	DINO	MoCo2+	NNCLR
$\mathcal{C}_1$	85.83	**89.64**	82.00	**83.77**	51.71	79.23
$\mathcal{C}_2$	42.58	53.44	42.94	59.04	29.33	46.89
$\mathcal{C}_3$	81.52	88.07	81.06	80.93	70.25	81.39
$\mathcal{C}_4$	**88.96**	84.20	**82.24**	83.36	**74.64**	**81.68**
$\mathcal{I}_1$	87.67	84.83	84.70	86.51	89.10	87.77
$\mathcal{I}_2$	66.76	60.84	68.33	66.18	63.61	63.18
$\mathcal{I}_3$	82.30	81.43	77.42	80.62	84.59	83.47
$\mathcal{I}_4$	**98.97**	**98.97**	**99.02**	**98.96**	**98.99**	**98.96**

Table 1. Different configurations of protected samples' ASR (%) are evaluated. $\mathcal{C}_1$–$\mathcal{C}_4$represent the downstream datasets CIFAR10, STL10, GTSRB, and ImageNet, respectively, with the encoder's pre-training dataset being CIFAR10. $\mathcal{I}_1 - \mathcal{I}_4$ use ImageNet as the pre-training dataset, and the downstream datasets are the same as those in $\mathcal{C}_1 - \mathcal{C}_4$. Barlow Twins and DeepCluster v2 are abbreviated as Barlow and DeepC2, respectively.

4.3. Ablation Study

In this section, we investigate the impact of using different datasets, modules, and attack intensities on RAEncoder when generating the protected dataset. We opt for pre-trained encoders on CIFAR10 and select CIFAR10 as the protected dataset for our analysis.

The Effect of Interference Pattern We investigate the impact of the interference pattern. Specifically, as shown in Figure 5a, we examine the cases where noise encoders were deployed with and without the interference pattern, and subjected to random noise inputs to simulate attacks where adversaries aim to steal correct adversarial perturbations δ, using random noise attacks. The results depicted in Figure 5a reveal that for noise encoders without interference pattern guidance, the attacked dataset obtained after being subjected to adversaries' random noise attacks has already lost its adversarial strength. However, noise encoders guided by the interference pattern exhibit robust adversarial strength even after undergoing attacks.



(a) (b)

Figure 5. Ablation study results of different settings. (a) Impact of deploying the interference pattern on the ASR (%) of the protected dataset and attacked dataset generated by the noise encoder. (b) The impact of different upper-bounds on attack performance and visual quality.

The Effect of ϵ We investigate the impact of four different perturbation upper-bounds ϵ on the attack performance and visual quality of protected samples. As shown in Figure 5b, different upper bounds influence both the attack performance and visual quality.

The Weights of Losses This experiment aims to study the mechanism of our method. From Eq. 9 to 17, we design six losses to train the noise encoder. As our target is to attack pre-trained encoder, the $\mathcal{L}_{E\_adv}$ is necessary, and we set its weight to 5 which is the default value as validated in [44].

To more intuitively assess imperceptibility, we introduced a new metric: Learned Perceptual Image Patch Similarity (LPIPS) [39] to evaluate the visual quality of the protected dataset. In contrast to SSIM and PSNR, LPIPS utilizes deep networks and is specifically designed to mimic human perception in image evaluation tasks. A higher LPIPS value indicates that the differences between images are more perceptible to human perception. Table 2 displays results from ablation experiments of $\mathcal{L}_{E\_\mathcal{H}}$ and $\mathcal{L}_{E\_mse}$, demonstrating their importance in enhancing the imperceptibility of adversarial samples.

To better evaluate the importance of the interference pattern, we conduct an ablation study on the values of the three losses introduced in the interference pattern: $\mathcal{L}_{I\_adv}$, $\mathcal{L}_{I\_\mathcal{H}}$, and $\mathcal{L}_{I\_mse}$. Specifically, we evaluate the performance of

$\mathcal{L}_{E\_mse}$	$\mathcal{L}_{E\_\mathcal{H}}$	SSIM	PSNR	LPIPS
✓	✗	0.80	28.22	0.194
✗	✓	0.82	28.24	0.165
✓	✓	**0.85**	**28.26**	**0.148**

Table 2. The effect of $\mathcal{L}_{E\_mse}$ and $\mathcal{L}_{E\_\mathcal{H}}$ on imperceptibility.

the protected dataset and the attacked dataset, recovered by the attacker using erroneous signature noise inputs to the noise encoder. We denote the ratio of the weights of $\mathcal{L}_{I\_adv}$, $\mathcal{L}_{I\_\mathcal{H}}$, and $\mathcal{L}_{I\_mse}$ to $\mathcal{L}_{E\_adv}$, $\mathcal{L}_{E\_\mathcal{H}}$, and $\mathcal{L}_{E\_mse}$ as ϕ, where $n : \alpha = m : \beta = \gamma : \lambda = \phi$. By controlling the value of ϕ, we can adjust the importance of the interference pattern in the learning process of the noise encoder. As shown in Table 3, when $\phi = 0.5$, the noise encoder's sensitivity to the interference pattern is reduced, leading to a decrease in $ASR_{Attacked}$, although the reduction is not substantial. Conversely, when $\phi = 2$, the noise encoder focuses more on learning the interference pattern, which increases the $ASR_{Attacked}$ but results in a reduced performance on the protected dataset. Thus, the weight of $\mathcal{L}_{I\_adv}$, $\mathcal{L}_{I\_\mathcal{H}}$ and $\mathcal{L}_{I\_mse}$ used to control the effect of the interference pattern are set to the same value as $\mathcal{L}_E$ to ensure that the learning of interference patterns is as important as the learning of normal patterns.

ϕ	$ASR_{Protected}$	$ASR_{Attacked}$	SSIM	PSNR
0.5	85.47	84.50	0.82	28.22
1(Default)	85.83	86.11	0.85	28.26
2	82.13	89.50	0.76	26.32

Table 3. The effect of ϕ on interference pattern.

4.4. Visualization Results

In this section, we compare the visual quality of protected and recovered samples generated by RAEncoder and existing methods. As shown in Figure 6, under conditions where the adversarial strength is superior, the protected samples produced by RAEncoder still maintain superior imperceptibility when compared to those generated by SRAE. As shown in Table 4, on CIFAR10, the protected dataset generated by RAEncoder achieves higher SSIM and PSNR values compared to the existing RDH-based method by [32] and the generative methods GEAA and SRAE. On ImageNet, the protected dataset generated by RAEncoder outperforms those by [32] and SRAE but is slightly weaker than PGN. However, PGN's visual quality cannot guarantee its adversarial strength which is explained in Section 4.5. Overall, RAEncoder demonstrates its superior imperceptibility, and the recovered dataset obtained through RAEncoder is identical to the original dataset. Importantly, these results are achieved without the need for an additional classification

head, but rather through direct utilization of a pre-trained encoder for the attack.

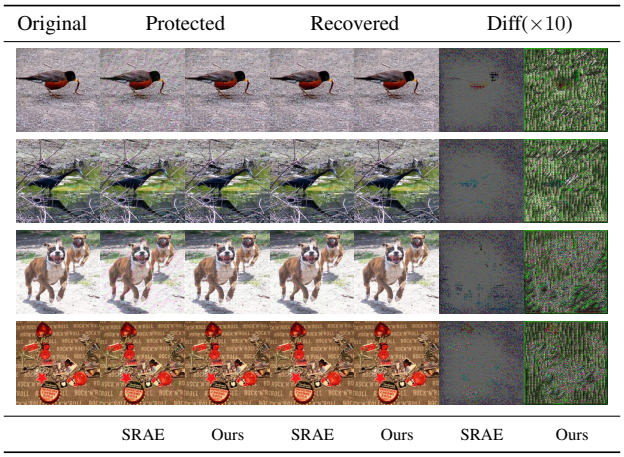


Figure 6. Visualization of the comparison between protected samples and recovered samples generated by our method and SRAE. Best viewed in zoom.

Dataset	Method	Protected		Recovered	
		SSIM	PSNR	SSIM	PSNR
CIFAR10	Xue *et al*. [32]	0.77	25.76	0.99	56.36
	GEAA [41]	0.32	27.91	0.99	48.11
	SRAE[38]	0.42	14.73	0.76	27.67
	Ours	**0.85**	**28.26**	**1.00**	∞
ImageNet	Xue *et al*. [32]	0.58	20.47	0.98	50.15
	PGN [31]	0.94	34.23	1.00	∞
	SRAE[38]	0.76	27.67	0.97	38.67
	Ours	0.82	30.26	1.00	∞

Table 4. Visual performance comparison of protected and recovered datasets from our method and existing methods.

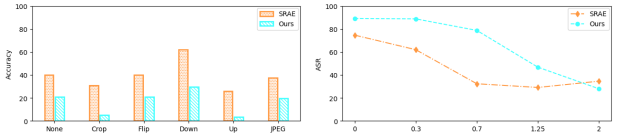
4.5. Transferability Study

In this section, we delve into the transferability of the RAEncoder from two distinct perspectives: attacks on downstream models trained with different pre-trained encoders, and attacks on various SL models with different network architectures. The downstream models used to evaluate transferability are trained on the corresponding pre-trained datasets of BYOL. The target SL model employed for [32], PGN and SRAE to generate the CIFAR10 protected dataset is ResNet18 and for ImageNet is ResNet50. High ASR is indicative of better adversarial strength and transferability. As shown in Table 5, the protected dataset generated by RAEncoder exhibits strong adversarial strength across various datasets, different SL models, and diverse pre-trained encoder configurations, resulting in significantly high ASR. Furthermore, its transferability across various SL models and pre-trained encoders frequently surpasses existing methods. However, while [32], PGN and SRAE demonstrate equally outstanding transferability across SL models, their ASR dramatically decreases when applied to downstream models trained by pre-trained encoders. It is noteworthy that, when applied to the CIFAR10 dataset, the ASR of SRAE is marginally higher than that of RAEncoder. However, as indicated in the Table 4, this increase in ASR is attained at the expense of image quality. Based on the results presented in Section 4.4 and Table 5, it is evident that our proposed RAEncoder effectively maintains the imperceptibility of protected samples while achieving a high level of adversarial strength.

4.6. Robustness Study

4.6.1. Image Processing Operation

We consider four image processing operations as attack measures from the perspective of attackers, aiming to mitigate the adversarial strength of protected samples. As depicted in Figure 7a, the protected samples generated by our proposed RAEncoder exhibit superior robustness compared to those generated by SRAE. Even after undergoing various image processing operations, the protected samples generated by RAEncoder maintain their adversarial effect, rendering the model incapable of correctly classifying the processed samples. This demonstrates that RAEncoder possesses enhanced robustness.



(a) (b)

Figure 7. The attack performance of RAEncoder in various settings. (a) From left to right, no disturbance, Center crop(proportion = 90%), Flip(horizontal and vertical), Resize(half of the height and width), Resize (twice the height and width) and JPEG compression(quality factor = 70). (b) Different degree of corruption.

4.6.2. Corruption

We take into consideration attackers may diminish the adversarial strength by introducing varying degrees of Gaussian blur to the samples, thereby degrading the images. As depicted in Figure 7b, with an increase in the degree of Gaussian perturbation, the ASR decreases. Particularly, when the blur degree reaches 1.25, the ASR of SRAE has dropped to approximately 30%, whereas our RAEncoder maintains an ASR of over 50%.

		VGG16	ResNet18	ResNet50	Dense121	Balow	BYOL	DeepC2	DINO	MoCo2+	NNCLR
Xue *et al.* [32]	CIFAR10	40.12	48.51	32.17	42.40	48.97	47.68	58.12	54.88	33.84	48.96
	ImageNet	73.04	88.18	76.42	64.28	74.16	87.56	56.64	0.60	9.56	48.44
PGN [31]	CIFAR10	-	-	-	-	-	-	-	-	-	-
	ImageNet	81.32	68.22	66.44	51.76	67.26	77.76	49.26	0.58	9.24	34.12
SRAE [38]	CIFAR10	61.84	88.40	76.79	72.47	73.68	88.82	90.45	51.65	80.03	80.44
	ImageNet	72.30	87.22	84.06	79.94	74.60	92.00	55.52	66.00	11.26	20.28
Ours	CIFAR10	58.79	84.79	70.66	**74.52**	**75.05**	**89.64**	89.79	**58.82**	**85.37**	**83.18**
	ImageNet	**98.22**	**91.46**	**88.30**	**99.00**	**89.04**	**98.97**	53.46	**86.00**	**30.84**	**51.80**

Table 5. Comparison study of ASR (%) of protected datasets across different SL models and pre-trained encoders. '-' denotes experiments not conducted.

4.6.3. Adaptive Attack

We consider the worst-case scenario for security concerns when deploying RAEncoder in practical applications (i.e., adaptive attack [32]). In such a scenario, attackers have full knowledge of the dataset protection methods, with the signature noise being the only unknown factor. The attacker is aware of the specifications of signature noise $\mathcal{S}$. However, due to the presence of the designed interference pattern, attackers are unable to reconstruct the original sample without signature noise $\mathcal{S}$. As shown in Figure 8, attackers are unable to recover the original sample by inputting random noise into the noise encoder and obtaining the output perturbation. Moreover, this even results in deeper levels of sample distortion.

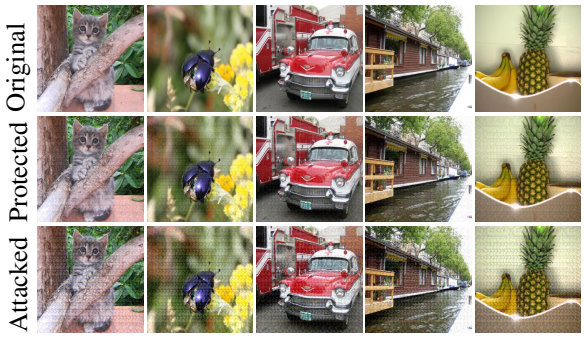


Figure 8. Examples of failure in image restoration by the attacker using incorrect signature noise.

4.7. Adversarial Strength Explanation

We aim to generate protected samples that are sufficiently distant from the original classes under small perturbations while remaining difficult to detect. As illustrated in Figure 9a and 9b, although PGN and SRAE can push some samples away from the original classes, the protected samples remain very close to the original samples in the feature space. However, as shown in Figure 9c, the method proposed by us, RAEncoder, yields protected samples that cluster together in the feature space, far from all original classes. Meanwhile, there are even cases where protected samples exhibit feature space almost identical to samples from other original classes, demonstrating the superior adversarial strength and imperceptibility of RAEncoder.

(a) PGN (b) SRAE (c) Ours

Figure 9. t-SNE [29] visualization of the feature space of protected examples produced by PGN, SRAE and ours. The hollow dots represent original samples and the solid dots represent RAEs.

5. Conclusion and Future work

In this paper, we highlight the novel challenges posed by the prevalence of SSL on RAEs for the first time. We introduce the first RAE framework applicable to protect the IP of unlabeled datasets. It leverages a noise encoder to generate universal adversarial perturbations for RAE generation, meeting the requirements of adversarial strength and full reversibility. Our RAEncoder exhibits significant performance improvement in all comparative experiments. We also evaluate its robustness against adaptive attacks, highlighting the practical advantages of RAEncoder over existing methods. Future work will include further exploration of balancing visual quality with adversarial strength. We also see new research opportunities in extending the attack approach against variational auto-encoders.

Acknowledgment. This work was supported in part by National Natural Science Foundation of China (NFSC, No. 62362025) and Hainan Province Science and Technology Special Fund (No. ZDYF2022GXJS224).

References

[1] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. Ieee, 2017. 2

[2] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020. 5

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 5

[4] Pengfei Chen, Leida Li, Haoliang Li, Jinjian Wu, Weisheng Dong, and Guangming Shi. Dynamic expert-knowledge ensemble for generalizable video quality assessment. *IEEE Trans. Circuits Syst. Video Technol.*, 33(6):2577–2589, 2023. 4

[5] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 5

[6] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9640–9649, 2021. 2

[7] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011. 5

[8] Victor Guilherme Turrisi Da Costa, Enrico Fini, Moin Nabi, Nicu Sebe, and Elisa Ricci. solo-learn: A library of self-supervised methods for visual representation learning. *Journal of Machine Learning Research*, 23(56):1–6, 2022. 5

[9] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9588–9597, 2021. 5

[10] Dahao Fu, Xiaoyi Zhou, Liaoran Xu, Kaiyue Hou, and Xianyi Chen. Robust reversible watermarking by fractional order zernike moments and pseudo-zernike moments. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(12):7310–7326, 2023. 1

[11] Mengxin Gong, Xiuli Chai, Yang Lu, and Yushu Zhang. Exploiting four-dimensional chaotic systems with dissipation and optimized logical operations for secure image compression and encryption. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(8):7628–7642, 2024. 1

[12] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020. 2, 5

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[14] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 2

[15] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 5

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. https://doi.org/10.48550/arXiv.1412.6980. 5

[17] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5

[18] Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC, 2018. 2

[19] Fengyong Li, Zhen Qi, Xinpeng Zhang, and Chuan Qin. Jpeg reversible data hiding using dynamic distortion optimizing with frequency priority reassignment. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(12):8849–8863, 2022. 1

[20] Jiayang Liu, Weiming Zhang, Kazuto Fukuchi, Youhei Akimoto, and Jun Sakuma. Unauthorized ai cannot recognize me: Reversible adversarial example. *Pattern Recognit.*, 134:109048, 2023. https://doi.org/10.1016/j.patcog.2022.109048. 1, 2

[21] Cheng Luo, Qinliang Lin, Weicheng Xie, Bizhu Wu, Jinheng Xie, and Linlin Shen. Frequency-driven imperceptible adversarial attack on semantic similarity. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15315–15324, 2022. 2

[22] Chengzhi Mao, Mia Chiquier, Hao Wang, Junfeng Yang, and Carl Vondrick. Adversarial attacks are reversible with natural supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 661–671, 2021. 2

[23] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016. 2

[24] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 4

[25] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 5

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[27] Kaiyu Song, Hanjiang Lai, Yan Pan, and Jian Yin. Mimicdiffusion: Purifying adversarial perturbation via mimicking clean diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24665–24674, 2024. 1

[28] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32:323–332, 2012. 5

[29] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008. 8

[30] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4): 600–612, 2004. https://doi.org/10.1109/TIP.2003.819861. 5

[31] Lizhi Xiong, Yue Wu, Peipeng Yu, and Yuhui Zheng. A black-box reversible adversarial example for authorizable recognition to shared images. *Pattern Recognit.*, 140: 109549, 2023. https://doi.org/10.1016/j.patcog.2023.109549. 1, 2, 3, 7, 8

[32] Mingfu Xue, Yinghao Wu, Yushu Zhang, Jian Wang, and Weiqiang Liu. Dataset authorization control: protect the intellectual property of dataset via reversible feature space adversarial examples. *Appl. Intell.*, 53(6):7298–7309, 2023. https://doi.org/10.1007/s10489-022-03926-1. 1, 2, 3, 4, 5, 6, 7, 8

[33] Zhaoxia Yin, Hua Wang, Li Chen, Jie Wang, and Weiming Zhang. Reversible adversarial attack based on reversible image transformation. *arXiv preprint arXiv:1911.02360*, 2019. https://doi.org/10.48550/arXiv.1911.02360. 3

[34] Zhaoxia Yin, Li Chen, Wanli Lyu, and Bin Luo. Reversible attack based on adversarial perturbation and reversible data hiding in yuv colorspace. *Pattern Recognit. Lett.*, 166:1–7, 2023. https://doi.org/10.1016/j.patrec.2022.12.018. 1, 2

[35] Jiang Yu, Jing Zhang, Zichi Wang, Fengyong Li, and Xinpeng Zhang. Cover selection in encrypted images. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2024. 1

[36] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning*, pages 12310–12320. PMLR, 2021. 5

[37] Haodong Zhang, Chi Man Pun, and Xia Du. Reversible adversarial examples with beam search attack and grayscale invariance. *arXiv preprint arXiv:2306.11322*, 2023. https://doi.org/10.48550/arXiv.2306.11322. 1, 2, 3

[38] Jiawei Zhang, Jinwei Wang, Hao Wang, and Xiangyang Luo. Self-recoverable adversarial examples: a new effective protection mechanism in social networks. *IEEE Trans. Circuits Syst. Video Technol.*, 33(2):562–574, 2022. https://doi.org/10.1109/TCSVT.2022.3207008. 1, 2, 3, 7, 8

[39] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6

[40] Weiming Zhang, Hui Wang, Dongdong Hou, and Nenghai Yu. Reversible data hiding in encrypted images by reversible image transformation. *IEEE Trans. Multimedia*, 18 (8):1469–1479, 2016. https://doi.org/10.1109/TMM.2016.2569497. 3

[41] Mengnan Zhao, Bo Wang, Wei Wang, Yuqiu Kong, Tianhang Zheng, and Kui Ren. Guided erasable adversarial attack (geaa) toward shared data protection. *IEEE Trans. Inf. Forensics Security*, 17:2468–2482, 2022. https://doi.org/10.1109/TIFS.2022.3186791. 1, 2, 3, 7

[42] Xiaoyi Zhou, Kaiyue Hou, Yujian Zhuang, Zhaoxia Yin, and Wenbao Han. General pairwise modification framework for reversible data hiding in jpeg images. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(1):153–167, 2024. 1

[43] Ziqi Zhou, Shengshan Hu, Minghui Li, Hangtao Zhang, Yechao Zhang, and Hai Jin. Advclip: Downstream-agnostic adversarial examples in multimodal contrastive learning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6311–6320, 2023. 1, 2

[44] Ziqi Zhou, Shengshan Hu, Ruizhi Zhao, Qian Wang, Leo Yu Zhang, Junhui Hou, and Hai Jin. Downstream-agnostic adversarial examples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4345–4355, 2023. 1, 2, 6