International Journal of Computer Vision
https://doi.org/10.1007/s11263-024-02140-5

Pattern-Expandable Image Copy Detection

Wenhao Wang[1] · **Yifan Sun**[2] · **Yi Yang**[3]

Received: 7 December 2023 / Accepted: 31 May 2024

Abstract
Open-world visual recognition aims to empower models to identify objects in real-world settings, particularly when they encounter domains or categories that are not included in the training dataset. This paper proposes a specific open-world visual recognition task, i.e. Pattern-Expandable Image Copy Detection (PE-ICD). In realistic scenarios, the continuous emergence of novel tampering patterns necessitates fast upgrades to the ICD system to prevent confusion in already-trained models. Therefore, our PE-ICD focuses on two aspects, i.e., rehearsal-free upgrade and backward-compatible deployment: (1) The rehearsal-free upgrade utilizes only the new patterns to save time, as re-training on the old patterns can be very time-consuming. (2) The backward-compatible deployment allows for comparing the updated query features against the outdated gallery features, thereby avoiding the need to re-extract features for the extensively large gallery. To lay the foundation for PE-ICD research, we construct the first regulated pattern set, *CrossPattern*, and propose Pattern Stripping (P-Strip). CrossPattern regulates both base and novel patterns during the initial training and subsequent upgrades. Given a query, our P-Strip separates the tamper patterns by decomposing it into an image feature and multiple pattern features. The advantage of P-Strip is that we can easily introduce new pattern features with minimal impact on the image feature and previously seen pattern features. Experimental results show that P-Strip supports both rehearsal-free upgrading and backward compatibility. Our code is publicly available at https://github.com/WangWenhao0716/PEICD.

Keywords Image copy detection · Novel patterns · Rehearsal-free upgrade · Backward compatibility

1 Introduction

The goal of open-world visual recognition is to equip models with the ability to deal with domains or categories in real-world scenarios that are not present in the training dataset. This paper focuses on a specific aspect of open-world visual recognition, i.e., Image Copy Detection (ICD) with particular attention to the realistic concern of novel tamper patterns. Basically, ICD aims to identify whether a query image is copied from the gallery after tampering with. This technique is crucial for improving the quality of content on the internet, preventing copyright infringement, and identifying pirated images. Although ICD methods try to employ a wide range of patterns for training, it is still infeasible to enumerate all possibilities. New tamper patterns are continuously emerging and can easily confuse an already-trained ICD model. Therefore, it is crucial to promptly update the ICD system whenever a novel pattern is detected.

A key challenge for the ICD system upgrading is the efficiency/latency problem because the fast reaction ability is critical (Zhong et al., 2022; Wang et al., 2023b). If we ignore the latency problem, a thorough solution should be retraining the ICD model with "seen + novel" patterns and then extracting all the "gallery + query" features. In this pipeline, two factors are particularly time-consuming: training with already-seen patterns and re-extracting gallery features. This is because the already-seen patterns are relatively abundant (compared with the novel patterns), and the gallery is tremendously large. If these two factors can be removed from the upgrading process, the latency problem would be well addressed.

In this paper, we formally introduce Pattern-Expandable Image Copy Detection (PE-ICD), aiming at efficiently upgrading an already-trained ICD model for newly-merged

Communicated by Zhun Zhong.

✉ Yi Yang
yangyics@zju.edu.cn

[1] ReLER, University of Technology Sydney, Sydney, Australia

[2] Baidu Inc, Beijing, China

[3] College of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang, China

Published online: 22 June 2024

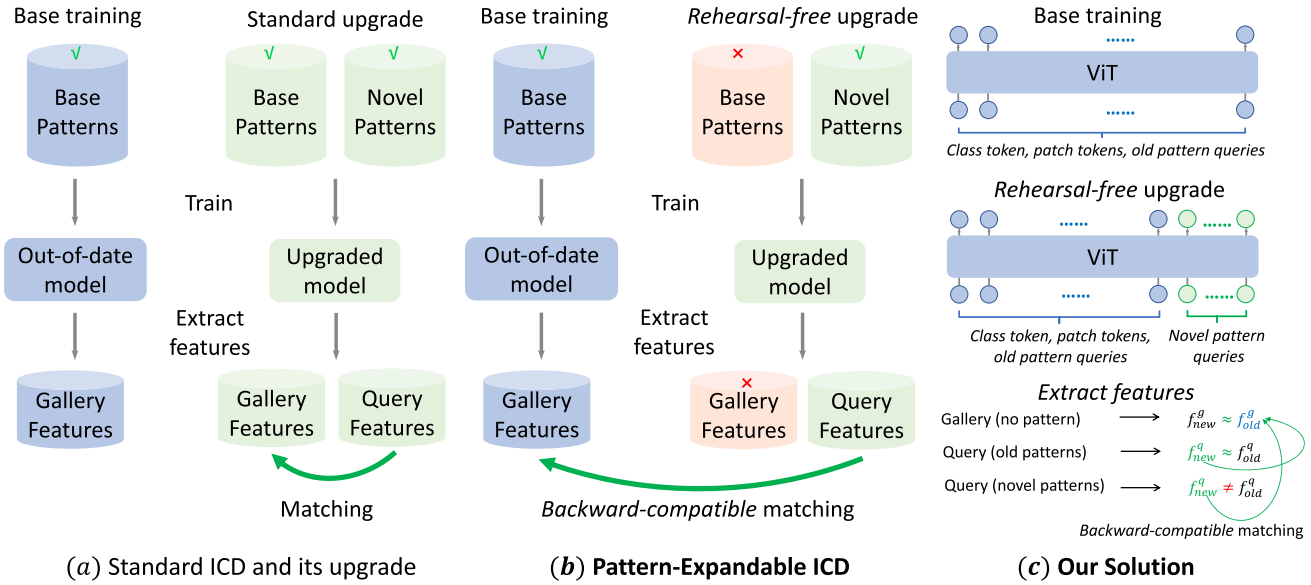


Fig. 1 Comparison between the standard ICD pipeline and the proposed PE-ICD pipeline with a corresponding solution. PE-ICD is featured for rehearsal-free upgrade and the backward-compatible matching. When novel tamper pattern emerges, PE-ICD only uses the novel patterns (no base patterns) to upgrade the ICD model. Moreover, PE-ICD only re-extracts the query features and compares the updated query features against the out-of-date gallery features. Rehearsal-free upgrade and backward compatibility avoid the cost of training on old patterns and inference on gallery features, respectively

tamper patterns. As shown in Fig. 1, compared with the standard ICD, PE-ICD has two emphases, i.e., rehearsal-free upgrade and backward-compatible deployment. (1) Rehearsal-free upgrade uses only the new patterns (NO old patterns) for upgrading the ICD models, because reusing the massive old patterns is very time-consuming. (2) Backward-compatible deployment means that features from the updated model are compatible to features from the out-of-date model. It enables direct comparison between the updated query feature and the outdated gallery feature, thereby avoiding the need to re-extract features for the significantly large gallery. Combining these two characteristics, PE-ICD removes the two most time-consuming factors for upgrading an ICD model and thus facilitates fast reaction against newly-merged tamper patterns.

To pave the way for PE-ICD research, we contribute the first regulated pattern set CrossPattern. This pattern set divides the tamper patterns into two sub-sets, one for base training and the other one for upgrade. Specifically, the base training only uses the base patterns, resulting in a base model that performs poorly on the novel patterns. Then the upgrading stage uses only the novel patterns to fine-tune the base model. During the inference, we use the out-of-date/upgraded model to extract the gallery/query features, respectively. We expect this upgrade (1) improves the ICD accuracy on the novel patterns (compared with base training) and (2) maintains the ICD accuracy on the base patterns. Specifically, we observe that the upper bound for ICD accuracy on novel patterns can be assessed through a standard upgrade (Fig. 1a), which is significantly more time-consuming than our PE-ICD (Fig. 1b). An effective PE-ICD should aim to narrow the gap toward this upper bound.

Moreover, we propose Pattern Stripping (P-Strip). Previous ICD methods, which learn a single feature for each image and make feature invariant to tamper patterns, are empirically found to lack rehearsal-free ability and backward-compatibility. In contrast, our P-Strip learns a image feature plus multiple tamper features. When comparing a query image (which might have been tampered with) against a gallery image, P-Strip first subtracts the tamper features from the query image feature and then compares the remaining feature against the gallery feature. In P-Strip, the impact of tampers is removed by the feature subtraction, yielding the so-called pattern stripping. As shown in Fig. 1c, we implement P-Strip with a Vision Transformer (ViT) (Dosovitskiy et al., 2020) backbone and use different queries for extracting different features from the shared backbone feature. The advantage of P-Strip is: it can easily incorporate features of novel patterns by adding new queries, while keeping the already-learned image feature and base pattern features unchanged. Our method is backward-compatible because: The gallery does not contain any patterns, resulting in minimal interaction with newly added pattern queries. Consequently, the gallery features remain nearly unchanged during the upgrade process. As a result, the updated query features, which are compatible with the updated gallery

features, are also compatible with the out-of-date gallery features. Experimental results also prove that P-Strip facilitates both rehearsal-free upgrading and backward compatibility.

To sum up, this paper makes the following contributions:

1. We propose a new ICD task called Pattern-Expandable Image Copy Detection (PE-ICD), which focuses on efficiently upgrading an already-trained ICD model for novel tamper patterns.
2. We build the first regulated pattern set, CrossPattern, and contribute Pattern Stripping (P-Strip) as a baseline method. P-Strip strips pattern features from an image feature and can be easily expanded for novel patterns based on the query mechanism of the transformer.
3. Extensive experimental results demonstrate the practicality of the proposed PE-ICD benchmark, as well as the rehearsal-free upgrading capability and backward compatibility of the proposed P-Strip method.

2 Related Works

2.1 Existing Image Copy Detection Methods

Past research has explored the problem of identifying copies or similar images through deep metric learning. For example, Multigrain (Berman et al., 2019) uses joint training to create image embeddings at multiple levels (i.e., class, instance, and copy). BoT (Wang et al., 2021b) provides a strong baseline for ICD with product-level matching accuracy using a 256-dimensional embedding. Another approach, called SSCD (Pizzi et al., 2022), adapts SimCLR (Chen et al., 2020) to the copy detection task by modifying the architecture and training objective. Recently, a study (Wang et al., 2023a) has focused on the hard-negative problem in ICD. However, these works employ various complex training patterns to cover all possible test scenarios, which may lead to overly optimistic results in their benchmarks. In contrast, this paper carefully regulates the patterns used for training and testing, taking the pattern-expansion problem seriously.

2.2 Image Retrieval

Image retrieval aims to return images that are similar or relevant to the query using various techniques, such as content-based image retrieval (Chaoyu et al., 2024; Rao et al., 2024; Zhu et al., 2023) and text-based image retrieval (Wang et al., 2023c; Choudhury et al., 2024; Wu et al., 2024). Our PE-ICD belongs to the category of content-based image retrieval. Beyond the general tasks of content-based image retrieval, we make some novel explorations: (1) The ICD task focuses on finer granularity: instead of considering the same object (Flusser et al., 2023) or instance (Chaoyu et al., 2024; Rao et al., 2024) as the true match, the true match in ICD is defined as edited copies or near-exact duplicates. (2) Unlike traditional image retrieval approaches that are generalizable or domain adaptive (Zhou et al., 2023; Li et al., 2023), we emphasize generalization across different patterns rather than images.

2.3 Compatible Representation Learning

Generally, compatible representation learning aims to generate new features compatible with existing ones. Some works, such as AML (Budnik & Avrithis, 2021), RBT (Wang et al., 2020) and CMP-NAS (Duggal et al., 2021), focus on the compatibility of features among different-sized models. BCT (Shen et al., 2020) proposes an influence loss that incorporates the learned classifier of the old embedding model into the training of the new embedding model to achieve backward compatibility in representation learning. Our PE-ICD demands backward compatibility and is thus closely related to compatible representation learning. However, there is a notable difference in terms of training. PE-ICD *only allows the upgrading of training data to only novel samples* (rehearsal-free setting), whereas compatible representation learning typically *allows the use of old data*.

2.4 Incremental Learning

Incremental learning involves training a model on a small initial dataset and then continuously updating the model as new data becomes available. This allows the model to learn and adapt to changes in the data distribution over time without the need to retrain on the entire dataset (Lao et al., 2023; Pu et al., 2023). Based on regularization techniques, LwF (Li & Hoiem, 2017) and EWC (Kirkpatrick et al., 2017) aim to prevent catastrophic forgetting by constraining the changes to the model's parameters during training. The iCaRL algorithm (Rebuffi et al., 2017) uses a combination of distillation and replay-based methods to retain previously learned information while training on new data. The updated model trained with incremental learning typically requires the gallery and query features update and thus does not consider backward compatibility. Therefore, these methods cannot be directly applied to the PE-ICD task.

3 Benchmark

This section first provides a brief overview of the publicly available ICD benchmarks and then elaborates on our PE-ICD benchmark.

3.1 Available Benchmarks

Currently, there are three publicly available ICD benchmarks, i.e. CopyDays (Douze et al., 2009), DISC21 (Douze et al., 2021), and NDEC (Wang et al., 2023a).

CopyDays (Douze et al., 2009) was introduced in 2009. It only provides 157 query images and 3,000 gallery images without training data. The tamper patterns are relatively simple, e.g., contrast changes and blurring.

DISC21 (Douze et al., 2021) is a comprehensive ICD benchmark proposed in 2021. It is designed for large-scale data, featuring one million training images and one million gallery images, and complex tamper patterns. Moreover, it includes many distractor queries, which have no true matches in the gallery.

NDEC (Wang et al., 2023a) considers the hard negative problem in ICD, i.e., some references are inherently similar to a query, but they are not copy-paste pairs. By featuring the hard-negative problem, NDEC makes ICD evaluation more realistic.

3.2 The Proposed PE-ICD Benchmark

We construct the first PE-ICD benchmark based on DISC21 images. Different from the original DISC21 dataset, our PE-ICD benchmark simulates the realistic scenario that new tamper patterns are emerging so that requires upgrading the out-of-date ICD system. Specifically, PE-ICD only allows 1) using base patterns for base training, and then 2) using novel patterns for upgrade. The test samples (query) have both "base + novel" patterns.

Definition and Evaluation Metric for PE-ICD. Formally, in the base training stage, the objective is to train a model g using only the base pattern set P_b. We denote the trained model at this stage as g_b. In the rehearsal-free upgrade, the PE-ICD allows upgrading the trained model g_τ^b with only the novel pattern set P_n. The updated model is denoted as $g_{b\to n}$ and is not allowed to extract reference features.

The query sets generated by the base and novel patterns are denoted as Q_b and Q_n, respectively. The reference set is denoted as R. PE-ICD is interested in (1) the extent to which the model forgets the base patterns after being trained on the novel one, i.e., the performance decline from $g_b\left(Q_b\right) \sim g_b\left(R\right)$ to $g_{b\to n}\left(Q_b\right) \sim g_b\left(R\right)$; and (2) the extent to which the model's performance improves on novel patterns after the upgrading stage, i.e., the performance improvement from $g_b\left(Q_n\right) \sim g_b\left(R\right)$ to $g_{b\to n}\left(Q_n\right) \sim g_b\left(R\right)$.

Therefore, in addition to the commonly used μAP metric (Douze et al., 2021), we introduce a new metric specifically for PE-ICD called *pattern gain*:

$$\left(\mu\mathrm{AP}_{g_{b\to n}}^{Q_n} - \mu\mathrm{AP}_{g_b}^{Q_n}\right) - \left(\mu\mathrm{AP}_{g_b}^{Q_b} - \mu\mathrm{AP}_{g_{b\to n}}^{Q_b}\right), \tag{1}$$

where $\mu\mathrm{AP}_g^Q$ is the μAP of model g tested on query set Q.

This protocol thoroughly evaluates both the performance decline on the base set of patterns and the performance enhancement on the novel set of patterns after upgrading.

Details of Our Benchmark. In PE-ICD, the model is first trained on base patterns are then upgraded on novel patterns. Correspondingly, the PE-ICD benchmark pre-defines a pattern set, $CrossPattern$, including two sets of training patterns. Specifically, the training set is divided into two groups with 11 base patterns and 11 novel patterns. The testing set uses all the 22 patterns to generate the query images. The base patterns include *RandomCrop*, *RandomRotate*, *HoriFlip*, *RandomBright*, *RandomContrast*, *RandomContrast*, *RandomOpacity*, *RandomEmoji*, *RandomImage*, *RandomPad*, *RandomPers*, *RandomPixel*, and *RandomShuffle*. The novel patterns include *RandomBlur*, *RandomSatur*, *RandomText*, *GrayScale*, *RandomMeme*, *RandomStripe*, *RandomNoise*, *RandomSharp*, *RandomSkew*, *VertFlip*, and *OverlayScreen*. See the details of all 22 patterns in the Appendix.

The original images (without tamper) are from the DISC21 dataset. Concretely, the training dataset is the same as the DISC21 training dataset, which contains 1 million unlabeled images and provides a sufficient source for adding patterns. The gallery dataset is also the same as the DISC21 gallery dataset. Finally, we build two query datasets using the two sets of patterns and evaluate models performance on these datasets to assess how well models perform on different patterns.

4 Method

In this section, we first present the preliminary and then introduce the proposed Pattern Stripping (P-Strip) method for the PE-ICD task.

4.1 Preliminary

4.1.1 The ICD Baseline

The ICD baseline (Wang et al., 2021a, b, 2023a) usually consists of two steps, i.e., generating copy-paste pairs and deep metric learning. Although these two steps can be merged by instantly generating copy-paste pairs for each training mini-batch, the online generation can lead to lower GPU utilization and training inefficiency. Therefore, we follow the popular two-step pipeline (Wang et al., 2021a, b, 2023a).

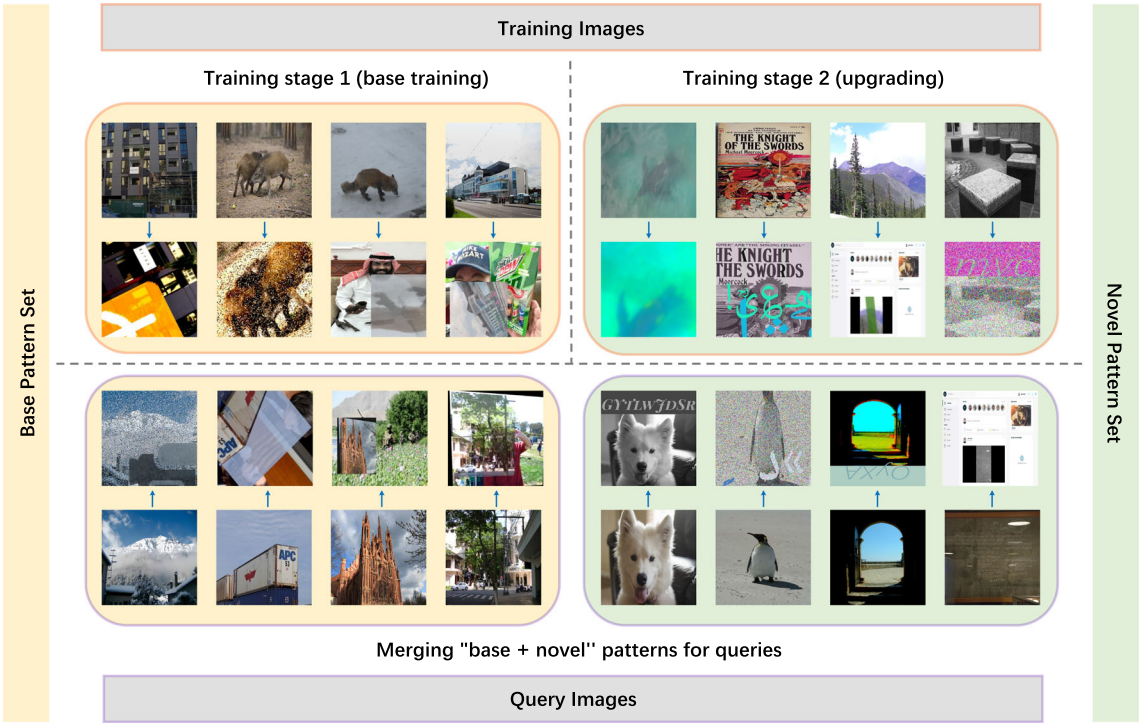
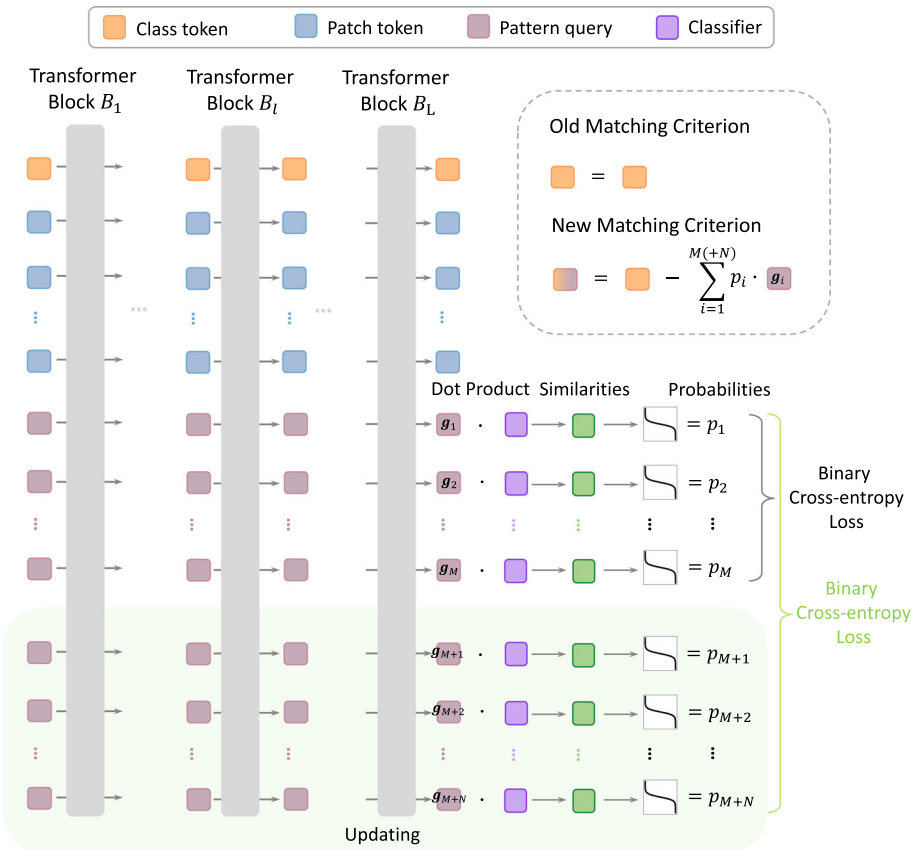


Fig. 2 The proposed PE-ICD benchmark. The images on the left and right are generated by different pattern sets. Therefore, there is a pattern gap between the two sets. The benchmark emphasizes the rehearsal-free upgrading and backward compatibility of models

Fig. 3 The illustration of Pattern Stripping (P-Strip). The process involves removing patterns from the feature level of an image. In this approach, pattern queries are trained as deep representations of patterns, and a class token is used to subtract the weighted sum of these queries to strip the patterns from the image. This method results in more efficient training and improved performance



Generating Copy-Paste Training Pairs. Given the unedited images, we use pre-defined patterns to generate a training dataset. Specifically, we randomly select several patterns from the holistic pattern set, use them to transform the image into some edited copies. As such, each edited copy and the original image form a positive copy-paste pair and thus share a "labeled" training class. In practice, for each unedited image, we generate multiple positive edited copies.

Deep Metric Learning on the Copy-Paste Pairs. After generating "labeled" training classes using various patterns, deep metric learning is performed on these classes to train a model for ICD tasks. This can be done using pairwise training, classification training, or a combination of both. Common loss functions for pairwise training include triplet loss (Hermans et al., 2017) and N-pair loss (Sohn, 2016). Classification training can be performed using loss functions such as Large-margin Softmax loss (Liu et al., 2016), Circle loss (Sun et al., 2020), and CosFace (Wang et al., 2018). In this study, we choose the popular CosFace (Wang et al., 2018), considering its effectiveness and simplicity.

4.1.2 Rehearsal-Free Upgrade

Definition and Mechanism. Rehearsal-free upgrade is a technique used in incremental learning or continual learning, where a model is required to learn new information or tasks over time while retaining its performance on previously learned tasks without explicit retraining on the old data. The rehearsal-free upgrade aims to efficiently adapt deep learning models to new information or tasks without the need for extensive retraining on old data, thereby overcoming the challenge of catastrophic forgetting.

Realization in ICD. In the context of ICD, the system is designed to learn new tampering patterns without the need to retrain on old patterns. This is achieved by introducing a mechanism that allows the model to adapt to new patterns efficiently while retaining its ability to detect previously learned patterns. This approach saves time and computational resources, as retraining on old patterns can be very time-consuming.

4.1.3 Backward-Compatible Deployment

Definition and Mechanism. Backward-compatible deployment ensures that an updated model can still function correctly and produce consistent results in an existing system, even if the system was designed for an older version of the model. This concept is crucial in scenarios where models are continuously updated or improved, but the systems relying on them cannot be updated simultaneously or frequently.

Realization in ICD. In the context of ICD, the updated model can compare new query features against outdated gallery features without the need to re-extract features for the entire gallery. This ensures that the system remains compatible with existing features, even after it has been updated to detect new tampering patterns.

4.2 Pattern Stripping (P-Strip)

The Intuition of P-Strip. Our method is featured for explicitly stripping pattern features from query features, so that novel tamper patterns can be easily added. We first explain the intuition of P-Strip in detail as below:

We note that popular methods (Berman et al., 2019; Wang et al., 2021b; Pizzi et al., 2022; Chen et al., 2020; Wang et al., 2023a) try to suppress / eliminate the impact of tamper patterns: Let us assume that an image x is transformed into an edited copy through $A(x)$. A popular ICD method learns a feature extractor $\mathbf{f}$ and tries to learn the pattern invariance with the below objective:

$$\mathbf{f}\left(A\left(x\right)\right) \mapsto \mathbf{f}\left(x\right), \tag{2}$$

where $\mapsto$ denotes approximating.

When updating $\mathbf{f}$ for novel tamper patterns, the training is prone to forgetting the already-learned knowledge on base tamper patterns (since we do not use base patterns during upgrading). In contrast, our P-Strip does not use feature extractor $\mathbf{f}$ for eliminating the variation caused by patterns. It disentangles the image feature and the pattern features and uses explicit feature subtraction operation to strip the impact of patterns, which is formulated as:

$$\mathbf{f}\left(A\left(x\right)\right) - \sum_{i=1}^{M} \mathbb{1}(i)\mathbf{g}_i\left(A(x)\right) \mapsto \mathbf{f}\left(x\right), \tag{3}$$

where $\mathbf{g}_i$ extracts the i-th pattern feature, M is the number of base patterns, $\mathbb{1}(i)$ indicates whether the i-th pattern exists in $A(x)$. In practice, we use predicted soft probability scores to replace $\mathbb{1}(i)$.

The advantage of P-Strip in Eq. 3 is: it can be expanded to accommodate novel patterns with little impact on the already learned image feature $\mathbf{f}$ and pattern feature $\mathbf{g}_i$. That is because: In the absence of P-Strip, image features inevitably contain pattern information. When novel patterns are introduced, the models struggle to adapt due to the existing gap between patterns. Consequently, these novel pattern features interfere with the feature extraction process, leading to less representative image features.

We formulate the upgrade training as:

$$\mathbf{f}\left(A\left(x\right)\right) - \sum_{i=1}^{M+N} \mathbb{1}(i)\mathbf{g}_i\left(A(x)\right) \mapsto \mathbf{f}\left(x\right), \tag{4}$$

where N is the number of novel patterns. It can be seen that learning novel pattern features $\mathbf{g}_i$ $(i = M+1, \cdots, M+N)$ is independent from the already learned features.

Implementing P-Strip with ViT and Query Mechanism. We use Vision Transformer (ViT) (Dosovitskiy et al., 2020) to implement the concept of P-Strip (Eqs. 3 and 4). The structure is illustrated in Fig. 3. For an input image, we follow ViT by tokenizing it and then feeding the tokens (class token + patch tokens) into the transformer. Meanwhile, we concatenate some multiple pattern queries (M for base training and $M+N$ for upgrading) as the context of the input tokens. These pattern queries pass through multiple ViT blocks, where they interact with image patch tokens via attention and detect the potential patterns present in the image.

In the base training stage, as shown in Fig. 3 (upper), we use the pattern queries in the last output layer as the patten features, i.e., $\mathbf{G} = [\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_M]$. $\mathbf{G}$ are then fed into a linear classifier parameterized with $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M]$. By comparing $\mathbf{g}_i$ to $\mathbf{w}_i$, we predict the probability score indicating whether the i-th pattern exists in the input image, which is formulated as:

$$p_i = \texttt{sigmoid}(\mathbf{w}_i^{\mathrm{T}} \cdot \mathbf{g}_i) = \frac{1}{1 + e^{-\mathbf{w}_i^{\mathrm{T}} \cdot \mathbf{g}_i}}. \tag{5}$$

The probability scores are supervised through the popular BCE loss as below:

$$\mathcal{L}_{bce} = \sum_{i=1}^{M} -(1 - y_i) \log(1 - p_i) - y_i \log(p_i), \tag{6}$$

where y_i is the ground-truth label ($y_i = 1$ if the i-th pattern exists).

We use the final class token **cls** as the image feature. Correspondingly, the P-Strip operation in Eq. 3 is implemented by:

$$\mathbf{cls} - \sum_{i=1}^{M} p_i \mathbf{g}_i. \tag{7}$$

During upgrading, as shown in Fig. 3 (lower), the pattern features are expanded to $\hat{\mathbf{G}} = [\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_{M+N}]$, and the parameterized linear classifier is expanded to

$$\hat{\mathbf{W}} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{M+N}]. \tag{8}$$

Corresponding, we have the expanded BCE loss:

$$\hat{\mathcal{L}_{bce}} = \sum_{i=1}^{M+N} -(1 - y_i) \log(1 - p_i) - y_i \log(p_i), \tag{9}$$

and the new P-Strip operation:

$$\mathbf{cls} - \sum_{i=1}^{M+N} p_i \mathbf{g}_i. \tag{10}$$

Training and Test Process. Considering that the PE-ICD benchmark requires the base training and upgrading stages, in the base training stage, only M (the number of patterns in the base pattern set) pattern queries are prepended. We add N new pattern queries to the trained model during the upgrading stage.

In the base training stage, all the parameters can be optimized and the training objective is

$$\mathcal{L} = \mathcal{L}_{mtr} + \lambda \cdot \mathcal{L}_{bce}, \tag{11}$$

where $\mathcal{L}_{mtr}$ is the metric learning loss (we use CosFace (Wang et al., 2018) here) and λ is the balance parameter.

In the upgrading stage, we use the same training objective, but to fulfill the backward compatibility,only the newly added N pattern queries and the classification head are trainable. This design avoids the destruction of the original trained model, leading to compatible features. However, having fewer trainable parameters makes training more difficult. Fortunately, our proposed P-Strip helps alleviate this problem, ultimately improving performance. Another advantage of this design is that when updating the embedding models stored in millions of computing nodes for testing, we only need to distribute the newly learned pattern queries, which results in a fast model upgrade process.

In the test stage, the model trained in the base training stage is used for the feature extraction of gallery images in two stages. The features of images in the two query sets are extracted by the models trained after the upgrading stage. The patterns contained in the query are predicted and stripped automatically. The rationales behind the design of re-extracting query features when reusing gallery features are: (1) The query may contain novel patterns that are not learned by the outdated model. However, the galleries consist of original images without tamper patterns. Therefore, we use an updated model to extract more discriminative features from the queries. This is the reason for updating our model. (2) In simulating a real-world scenario, we replace the original model with an updated one. In this situation, the outdated model is no longer used. For any new query (regardless of whether it contains base or novel patterns), we use the updated model to extract its features and compare them with the features of the outdated gallery.

5 Experiments

5.1 Training Details

We use PyTorch (Paszke et al., 2019) to implement our P-Strip method. It is trained on four Nvidia A100 GPUs. Unless otherwise stated, the backbone is ViT-B/16 (Dosovitskiy et al., 2020) that was pre-trained on the ImageNet dataset (Deng et al., 2009) using DeiT (Touvron et al., 2021). The images are resized to 224×224 pixels before training. We use a balance parameter λ of 0.5 and a batch size of 128. The standard PK sampling method is used in each batch, with 32 classes and 4 images per class. The number of epochs is 25, and we use a cosine-decreasing learning rate.

5.2 The Challenge from Novel Patterns

This section shows that a pattern gap exists and causes significant performance issues. We reimplement five state-of-the-art ICD algorithms (DINO (Caron et al., 2021), MultiGrain (Berman et al., 2019), SSCD (Pizzi et al., 2022), BoT (Wang et al., 2021b), and ASL (Wang et al., 2023a)) plus the baseline in this paper by regulating the patterns they are trained on to the base pattern set in the PE-ICD benchmark and evaluating them on the query sets generated by the two pattern sets. We also unify their backbones into ViT-B/16 to enable a fair comparison across different methods. From Table 1, we draw three observations: (1) Most methods perform well (around 90% μAP) when trained and tested on the base pattern set, indicating that they successfully learn invariance relative to patterns and generalize well to new images. (2) All methods experience a significant drop in performance (around -50% μAP) when tested in a novel-pattern setting, showing that all methods struggle in this practice setting, and the pattern gap remains an unsolved problem. (3) Our ICD baseline is strong enough without bells and whistles: in the base-pattern setting, we only perform -0.54% μAP compared to the strongest competitor (BoT); in the novel-pattern setting, we outperform all state-of-the-art methods. By proposing the PE-ICD benchmark to challenge the state-of-the-art methods, we call for more research efforts on the pattern-expansion problem.

Table 1 Evaluation in the base-pattern and novel-pattern settings. While most models perform well in the base-pattern setting, they all experience a significant drop in accuracy when tested on novel patterns

Method	μAP (%) (Base) ↑	μAP (%) (Novel) ↑
DINO	46.95	13.69
MultiGrain	73.35	24.01
SSCD	84.65	30.47
BoT	93.56	36.29
ASL	90.62	39.69
Our baseline	93.02	39.87

5.3 PE-ICD Increases the Upgrade and Deployment Efficiency

As shown in Table 2, our PE-ICD significantly reduces the costs associated with both upgrading models and re-extracting the gallery. (1) Training: assume there are n_b base and n_n novel patterns. Compared with the complete update (rehearsal), PE-ICD is rehearsal-free and only requires $\frac{n_n}{n_n+n_b}$ training time. In our experiments ($n_b = n_n = 11$), a complete update requires about 61 GPU hours, while PE-ICD requires only 31 GPU hours. (2) Test: due to the backward compatibility, PE-ICD does not re-extract the gallery features and saves 0.82 GPU hour for one million images with one A100 GPU. Our computational economy will be more significant for realistic large-scale datasets.

5.4 The Effectiveness of P-Strip

Our P-Strip Outperforms Other Plausible Approaches. As shown in Table 2 and Fig. 4, P-Strip is compared with two common strategies: fine-tuning and incremental learning. Notably, previous backward-compatible methods such as BCT (Shen et al., 2020) lack the capability to operate without old data and fail to converge in our PE-ICD context. In the fine-tuning approach, a model initially trained on a base pattern set is subsequently tuned with all parameters being trainable on a novel pattern set. For incremental learning, we adapt Learning without Forgetting (LwF) (Li & Hoiem, 2017) by using both updated and old models to extract features from the novel training set images and regulate the L_2 distance between these feature sets to prevent forgetting. Both fine-tuning and incremental learning are proved to be ineffective, showing a decrease in pattern gain by -44.53% and -26.89% respectively, due to significant model shifts. Furthermore, these methods require more time for updates compared to our P-Strip. Our approach effectively narrows the performance gap towards the upper benchmark set by a standard upgrade.

P-Strip Achieves Good Backward Compatibility. We evaluate the backward compatibility of P-Strip by comparing it with a backfilling counterpart in Table 3. Recall that P-Strip is free of backfilling: the upgraded model is used only to extract the query features, leaving the gallery features outdated. In contrast, the backfilling counterpart updates both the query and gallery features, which is time-consuming but ideally resolves the backward compatibility issue. We observe that: (1) On the base pattern test images, maintaining unchanged gallery features shows slightly better performance (about 1%). We speculate that this is because

Table 2 A comparison of our method to fine-tuning and incremental learning under a backfilling-free setting

Method	μAP (%) (Base) ↑	μAP (%) (Novel) ↑	Pattern gain (%) ↑	Upgrading cost	Re-extracting cost
Basl. (before upgrade)	93.02	39.87	–	–	–
Fine-tuning	30.94	57.42	−44.53	40 h	0
Incremental learning	51.36	54.64	−26.89	63 h	0
P-Strip	90.68	73.32	31.11	31 h	0
Upper bound	92.91	93.81	53.83	61 h	0.82 h

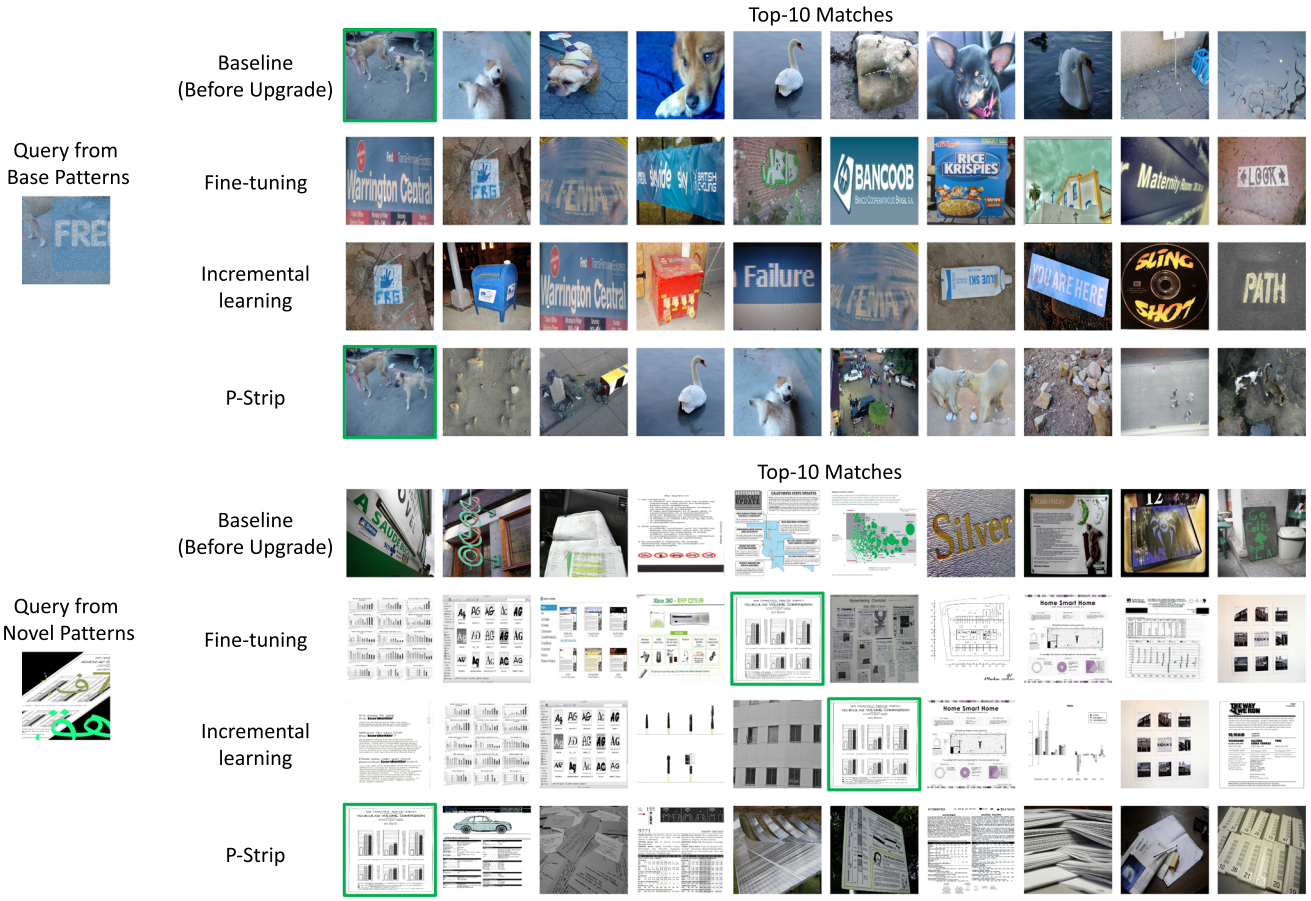


Fig. 4 Visual comparison between our proposed P-Strip and other methods. The original image (true match) is highlighted in green. Our P-Strip accurately identifies the original image, regardless of whether the query is generated by base or novel patterns

Table 3 The comparison between backfilling-free and backfilling settings

Methods	μAP (%) on base patterns		μAP (%) on novel patterns		Pattern gain
	Before upgrade	After upgrade	Before upgrade	After upgrade	
P-Strip (Backfilling-free)	93.02	90.68	39.87	73.32	31.11
P-Strip (Backfilling)	93.02	88.99	39.87	77.26	33.36

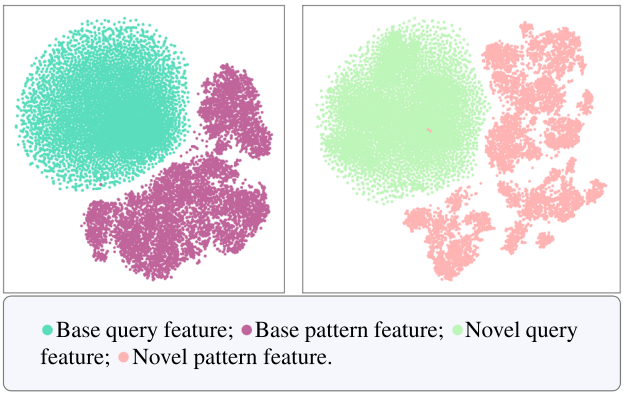


Fig. 5 The t-SNE (Van der Maaten & Hinton, 2008) visualization of feature distributions of image and pattern features. From the distribution, we conclude that the image and pattern features are different

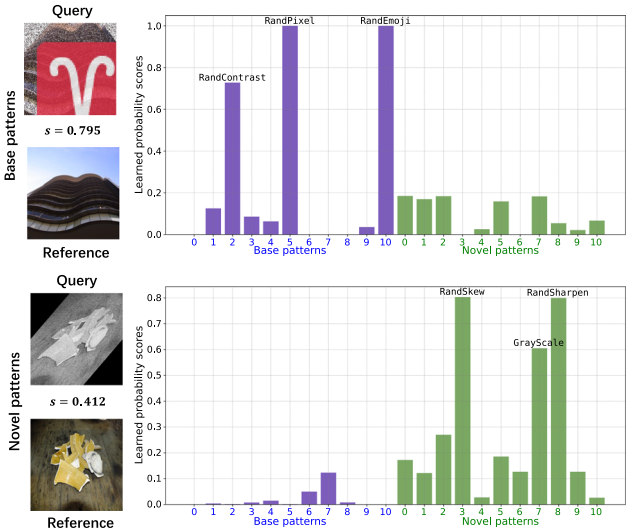


Fig. 6 The learned probability scores for queries generated by applying base and novel patterns, respectively

the gallery (true-original image) features extracted by the tuned model are slightly inferior. (2) On the novel pattern test images, the backfilling-free setting leads to only about a 4% μAP performance drop. This is reasonable because, even though we minimize changes to the model, the outdated features are still somewhat incompatible with the updated ones. (3) Overall, the performance gains in the two settings are comparable, indicating that our method is feasible by only updating the query features.

Direct Evidence of the Separability Between Image and Pattern Features. Remember that we denote an image as x and its edited copy as $A(x)$, and thus their images features are $\mathbf{f}(x)$ and $\mathbf{f}(A(x))$, respectively. The pattern feature of $A(x)$ is $\sum_{i=1}^{M} p_i \mathbf{g}_i\,(A(x))$ (as shown in Eq. 7). If the image and pattern features are separable, we should have:

$$\mathbf{f}\,(A\,(x)) \approx \mathbf{f}\,(x) + \sum_{i=1}^{M} p_i \mathbf{g}_i\,(A(x))\,, \tag{12}$$

which means the image feature of edited copy $A(x)$ can be represented by the sum of the image feature of image x and the pattern feature of pattern A.

To provide the direct evidence of the separability between image and pattern features, we calculate the cosine similarity s between the subtracted feature and the weighted sum of pattern features, i.e.,

$$s_b = < \mathbf{f}\,(A_b\,(x)) - \mathbf{f}\,(x)\,,\, \sum_{i=1}^{M} p_i \mathbf{g}_i\,(A_b(x)) >, \tag{13}$$

and

$$s_n = < \mathbf{f}\,(A_n\,(x)) - \mathbf{f}\,(x)\,,\, \sum_{i=1}^{M+N} p_i \mathbf{g}_i\,(A_n(x)) >, \tag{14}$$

where A_b and A_n represent the base and novel patterns, respectively; M and N denote the number of base and novel patterns, respectively.

Experimentally, the average values of s_b and s_n across the test set are 0.77 and 0.69, respectively. It thus (1) validates the separability of image and pattern features and (2) explains why μAP on novel patterns is relatively lower.

In addition to this evidence, we visualize the distribution of image and pattern features using t-SNE (Van der Maaten & Hinton, 2008), as illustrated in Fig. 5. From this analysis, we also conclude that the image and pattern features are separable (different) regardless of whether they are in the base or novel set.

Our Method Resists Error Accumulation Strongly in the Continuous Update Setting. We show that the proposed method resists error accumulation strongly by setting up a continuous update. Specifically, we partition the 11 novel patterns into 3 groups (3+4+4) for continuous model upgrades. The μAP on the base and first three novel patterns undergo 93.0% $\rightarrow$ 88.7% $\rightarrow$ 90.7% $\rightarrow$ 90.4% and 48.1% $\rightarrow$ 76.9% $\rightarrow$ 72.7% $\rightarrow$ 73.0%, respectively. We observe that: (1) For the base patterns, though the first updating (93.0% $\rightarrow$ 88.7%) brings unavoidable performance drop, the upcoming updating (88.7% $\rightarrow$ 90.7% $\rightarrow$ 90.4%) does not further damage the performance. (2) For the first three novel patterns, the first updating improves the performance significantly (48.1% $\rightarrow$ 76.9%). Similar to the base patterns, the next updating (with 4 different novel patterns) brings a little performance drop and continuous updating does not further damage performance (76.9% $\rightarrow$ 72.7% $\rightarrow$ 73.0%).

Visualization of the Learned Probability Scores. In Fig. 6, we visualize the learned probability scores, i.e. p_i in Eq. 5, for queries generated by base and novel patterns. The

Table 4 The ablation studies of our proposed P-Strip

Method	μAP (%) (Base) ↑	μAP (%) (Novel) ↑	pattern gain (%) ↑
Baseline (before upgrade)	93.02	39.87	–
Without BCE	90.48	58.74	16.33
With BCE	90.53	63.79	21.43
Meaningless stripping	90.80	64.12	22.03
Gallery without stripping	**90.94**	66.72	24.77
Pattern stripping	**90.94**	**66.73**	**24.78**

The best performance is indicated in bold

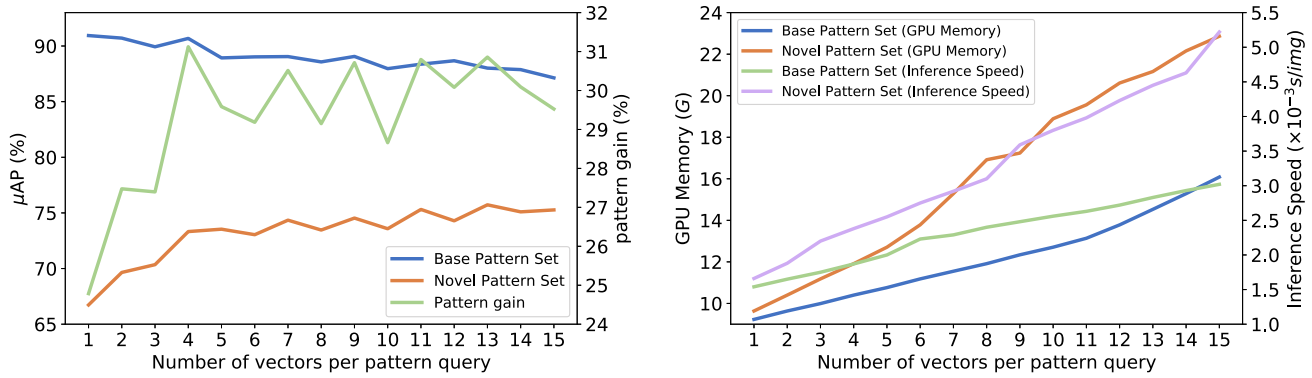


Fig. 7 The performance (left) and efficiency (right) analysis of using more vectors to represent a pattern query. More vectors lead to improved performance within a certain range, but the inference speed slows down, and the GPU memory usage increases

patterns with the top three highest scores are noted in the bar charts. We observe that: (1) Our method accurately predicts both base and novel patterns, demonstrating the effectiveness of the pattern stripping method in subtracting the expected pattern features. (2) The predictions for the base patterns are generally more accurate than those for the novel ones. In the second bar chart, for the 2nd novel pattern (`RandText`), the prediction probability score is near 0.3, even though it is evident that the query contains no text; while in the first bar chart, all of the probability scores for negative patterns are lower than 0.2.

5.5 Ablation Studies and Parameter Analysis

In this section, we first evaluate the effectiveness of using BCE loss and the performance improvement resulting from the stripping operation in Table 4. Then, in Fig. 7, we use more vectors to represent a pattern query and analyze the performance improvement and efficiency burden of this variant of our model.

The Effectiveness of Using BCE Loss. In Table 4, we show the effectiveness of using BCE loss by comparing the performance of the "Without BCE" and "With BCE" settings. In the "Without BCE" setting, we simply prepend M pattern queries to the input of ViT during training. These queries are trained or fine-tuned in the base training or upgrading stage, but without supervision, they are essentially meaningless. In the "With BCE" setting, we add the BCE loss to the training process but still omit the subtraction operation. From Table 4, we can see that: (1) Compared to the baseline (before upgrade), the naive approach of adding queries ("Without BCE") still leads to improved performance (+18.87% μAP) on the novel pattern set. This is because the added queries can be considered extra parameters that help the model fit the novel training set. (2) Compared to the naive approach, using the BCE loss for supervision (even without the subtraction operation) leads to much better performance (+5.05% μAP and +5.10% pattern gain). This is because allowing the transformer blocks to identify the patterns in an image helps guide the training process, resulting in a better fit.

The Performance Improvement Resulting from the Stripping Operation. We consider two variants of the pattern stripping: "meaningless stripping" and "gallery without stripping". "Pattern meaningless stripping" refers to when stripping the pattern feature from the image feature, we do not use the weighted sum (Eq. 7); instead, we subtract the sum of all pattern queries ($\sum_{i=1}^{M} \mathbf{g}_i$). The setting is used to prove that it is the "stripping the pattern feature of an image" rather than the "stripping operation itself" that works. Compared to Line 3 with Line 4 of Table 4, this variant barely brings any performance improvement. "Gallery without stripping" refers to when we extract the feature of galley images, we use the class token rather than the stripped class token as the deep representation. It is reasonable that this variant has no

performance difference from the “pattern stripping" because no pattern exists in the gallery images, proving our method’s correctness.

Using More Vectors to Represent a Pattern Query Further Improves Performance. In the method section, we only use one vector to represent a pattern query. However, there are other choices, such as using several vectors as one pattern query, and each vector is supervised as before. More vectors mean more trainable parameters in the upgrading stage. We show the performance and efficiency in Fig. 7. We draw the following conclusions: (1) Within a certain range ($1 \sim 4$ vectors), more vectors bring better performance: the μAP on the novel pattern is improved from 66.73% to 73.32%, and the pattern gain is improved from 24.78% to 31.11%. However, when the number of vectors exceeds 4, more vectors are useless, and the μAP and pattern gain fluctuate without increasing. This is because a greater number of vectors introduce more learnable parameters, which can better fit the training images generated by novel patterns. However, there is a limitation: beyond a certain point, performance begins to fluctuate, and no further improvement is observed. (2) More vectors increase the computing burden. When using the upgraded model, the occupied GPU memory of 15 vectors is about twice as much as using 1 vector. Also, the memory gap between using models trained on different stages increases as the number of vectors grows. The inference speed is also up to twice slower. Combining these two figures, we suggest that if more performance is needed and lower efficiency is bearable, using 4 vectors to represent a pattern query is a good choice. (3) The results on the base set always degrade. The reason is that although our method only learns new pattern queries and keeps the backbone frozen, during the feed-forward process, the class token and patch tokens of images generated by the base patterns inevitably attend to the new pattern queries and are thus disturbed.

6 Conclusion

This paper introduces the Pattern-Expandable Image Copy Detection (PE-ICD) task. Compared with the basic ICD, PE-ICD focuses on efficiently upgrading an out-of-date ICD system for additional novel tamper patterns. Two prerequisites for PE-ICD are using only the novel patterns (NO base patterns) for upgrade and backward compatibility, which are critical for fast reaction against novel tamper patterns. We contribute the first PE-ICD task, benchmark it with popular ICD methods, and proposes a strong baseline named Pattern-Stripping (P-Strip) method. P-Strip uses the vision transformer to stripe the pattern features from the query feature. Using the transformer query mechanism, P-Strip can learn to add novel pattern features with little impact on the query feature and already-seen pattern features. It thus facilitates efficient upgrading and good backward compatibility. We hope this work will draw research attention to a critical realistic problem, i.e., a fast reaction against novel tamper patterns, for the ICD system. In the future, we plan to explore more efficient ICD upgrading methods and multi-modal large language models for ICD.

Limitations and Social Impacts. PE-ICD is a very challenging new task. Using the proposed P-Strip, the ICD accuracy is significantly improved but is still much lower than the accuracy on base patterns. It indicates that the upgrading effect still has a large gap toward realistic application. Our research has positive social impacts because ICD is valuable for preventing copyright infringement.

Appendix A Demonstration of the Base and Novel Patterns

Tables 5, 6, and 7 display the names, detailed elaborations, and demonstrations of base and novel patterns. Although we use four samples to illustrate, in our PEICD, the query images have no overlap with the training images, a basic requirement for image retrieval tasks.

Table 5 The demonstration of the base and novel patterns (part 1) in our CrossPattern

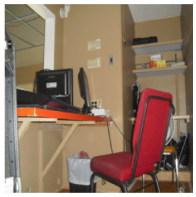
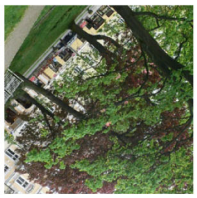
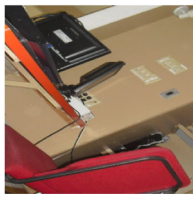
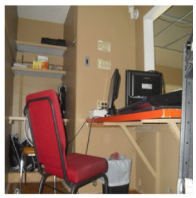
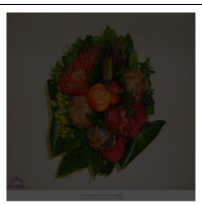
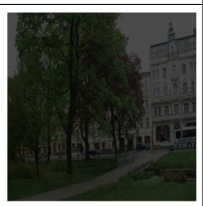
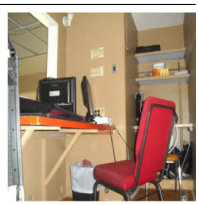
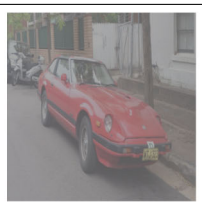
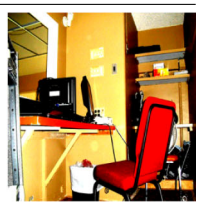
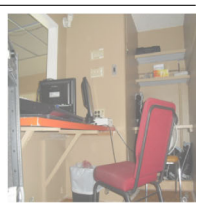
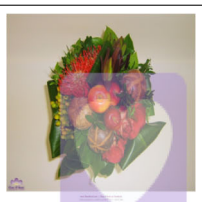
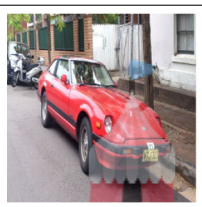
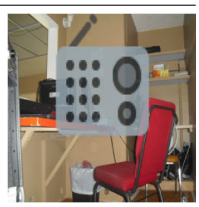
Pattern	Class	Elaboration	Demo 1	Demo 2	Demo 3	Demo 4
-	Origin	The samples to add patterns.				
Random Crop	Both	The common randomly resize and crop are base and novel patterns.				
Random Rotate	Base 0	Randomly rotate an image to a specified degree.				
Hori Flip	Base 1	Perform a horizontal flip of an image.				
Random Bright	Base 2	Randomly alter the brightness of an image.				
Random Contrast	Base 3	Adjust the contrast of an image randomly within a specified range.				
Random Opacity	Base 4	Randomly alter the transparency or opacity of an image.				
Random Emoji	Base 5	Randomly place an emoji onto an image.				

Table 6 The demonstration of the base and novel patterns (part 2) in our CrossPattern

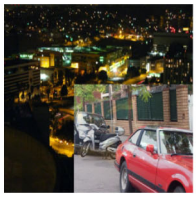
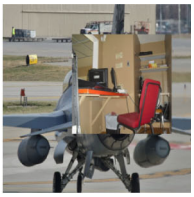
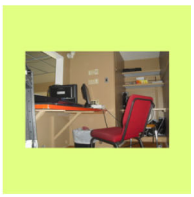
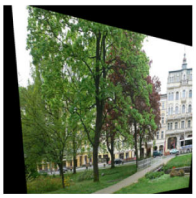
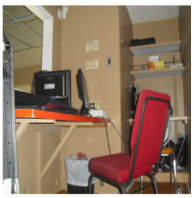
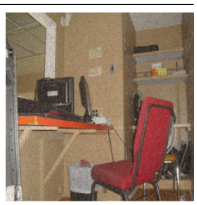
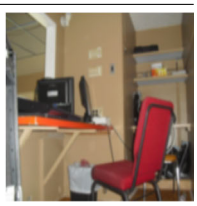
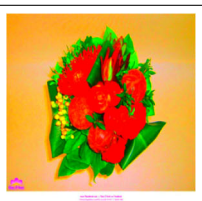
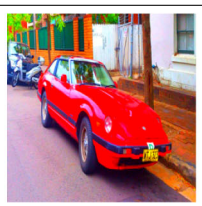
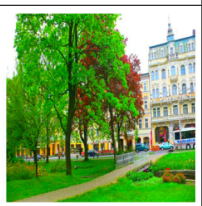
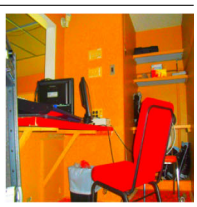
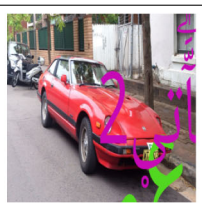
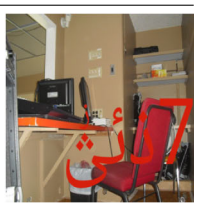
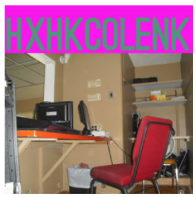
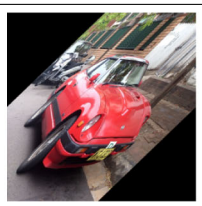
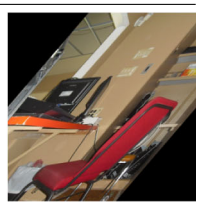
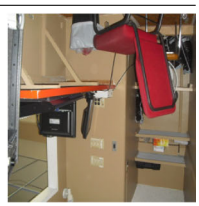
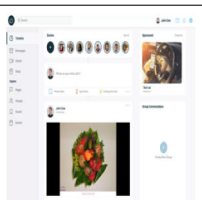
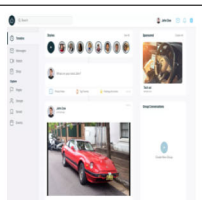
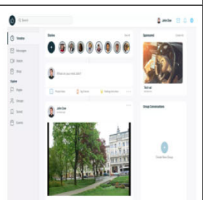
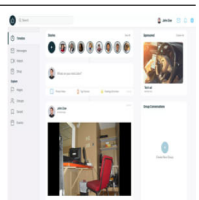
Pattern	Class	Elaboration	Demo 1	Demo 2	Demo 3	Demo 4
Random Image	Base 6	Overlay a randomly chosen image onto an image.				
Random Pad	Base 7	Add random padding to an image.				
Random Pers	Base 8	Apply a random perspective transformation to an image.				
Random Pixel	Base 9	Transform an image by randomly altering the resolution of its pixels.				
Random Shuffle	Base 10	Rearrange/Shuffle the pixels within an image randomly.				
Random Blur	Novel 0	Apply a random amount of blur to an image.				
Random Satur	Novel 1	Randomly adjust the color saturation of an image.				
Random Text	Novel 2	Place random texts over an image.				

Table 7 The demonstration of the base and novel patterns (part 3) in our CrossPattern

Pattern	Class	Elaboration	Demo 1	Demo 2	Demo 3	Demo 4
Gray Scale	Novel 3	Convert an image to grayscale.				
Random Meme	Novel 4	Randomly add a meme (a piece of text that is humorous in nature) to an image.				
Random Stripe	Novel 5	Apply randomly placed and randomly sized stripes over an image.				
Random Noise	Novel 6	Introduce random noise to an image.				
Random Sharp	Novel 7	Enhance the details in an image by sharpening.				
Random Skew	Novel 8	Apply a random skew transformation to an image.				
Vert Flip	Novel 9	Perform a vertical flip of an image.				
Overlay Screen	Novel 10	Simulate the appearance of the overlayed image on the screenshot.				

References

Berman, M., Jégou, H., Vedaldi, A., Kokkinos, I., & Douze, M. (2019). Multigrain: A unified image embedding for classes and instances. arXiv preprint arXiv:1902.05509

Budnik, M., & Avrithis, Y. (2021). Asymmetric metric learning for knowledge transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 8228–8238).

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., & Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

Chaoyu, Z., Jianjun, Q., Shumin, Z., Jin, X., & Yang, J. (2024). Learning robust facial representation from the view of diversity and closeness. *International Journal of Computer Vision, 132*(2), 410–427.

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning, PMLR* (pp. 1597–1607).

Choudhury, S., Laina, I., Rupprecht, C., & Vedaldi, A. (2024). The curious layperson: Fine-grained image recognition without expert labels. *International Journal of Computer Vision, 132*(2), 537–554.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 248–255). IEEE

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Douze, M., Jégou, H., Sandhawalia, H., Amsaleg, L., & Schmid, C. (2009). Evaluation of gist descriptors for web-scale image search. In *Proceedings of the ACM International Conference on Image and Video Retrieval* (pp. 1–8)

Douze, M., Tolias, G., Pizzi, E., Papakipos, Z., Chanussot, L., Radenovic, F., Jenicek, T., Maximov, M., Leal-Taixé, L., Elezi, I., et al. (2021). The 2021 image similarity dataset and challenge. arXiv preprint arXiv:2106.09672

Duggal, R., Zhou, H., Yang, S., Xiong, Y., Xia, W., Tu, Z., & Soatto, S. (2021). Compatibility-aware heterogeneous visual search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 10723–10732).

Flusser, J., Lébl, M., Šroubek, F., Pedone, M., & Kostková, J. (2023). Blur invariants for image recognition. *International Journal of Computer Vision, 131*(9), 2298–2315.

Hermans, A., Beyer, L., & Leibe, B. (2017). In defense of the triplet loss for person re-identification. arXiv preprint arXiv:1703.07737

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.

Lao, M., Pu, N., Liu, Y., Zhong, Z., Bakker, E.M., Sebe, N., & Lew, M.S. (2023). Multi-domain lifelong visual question answering via self-critical distillation. In *Proceedings of the 31st ACM International Conference on Multimedia, Association for Computing Machinery, New York, NY, USA* (pp. 4747–4758).

Li, W.H., Liu, X., & Bilen, H. (2023). Universal representations: A unified look at multiple task and domain learning. *International Journal of Computer Vision*, 1–25.

Li, Z., & Hoiem, D. (2017). Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 40*(12), 2935–2947.

Liu, W., Wen, Y., Yu, Z., & Yang, M. (2016). Large-margin softmax loss for convolutional neural networks. In *ICML*.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., & Lin, Z., et al. (2019). Pytorch: An imperative style, high-performance deep learning library.

Pizzi, E., Roy, S.D., Ravindra, S.N., Goyal, P., & Douze, M. (2022). A self-supervised descriptor for image copy detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 14532–14542).

Pu, N., Zhong, Z., Sebe, N., & Lew, M. S. (2023). A memorizing and generalizing framework for lifelong person re-identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 45*(11), 13567–13585.

Rao, H., Leung, C., & Miao, C. (2024). Hierarchical skeleton meta-prototype contrastive learning with hard skeleton mining for unsupervised person re-identification. *International Journal of Computer Vision, 132*(1), 238–260.

Rebuffi, S.A., Kolesnikov, A., Sperl, G., & Lampert, C.H. (2017). ICARL: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition* (pp. 2001–2010).

Shen, Y., Xiong, Y., Xia, W., & Soatto, S. (2020). Towards backward-compatible representation learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 6368–6377).

Sohn, K. (2016). Improved deep metric learning with multi-class n-pair loss objective. In *Advances in neural information processing systems* (Vol. 29).

Sun, Y., Cheng, C., Zhang, Y., Zhang, C., Zheng, L., Wang, Z., & Wei, Y. (2020). Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 6398–6407).

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. (2021). Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning, PMLR* (pp. 10347–10357).

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research, 9*(11).

Wang, C.Y., Chang, Y.L., Yang, S.T., Chen, D., & Lai, S.H. (2020). Unified representation learning for cross model compatibility. In *British Machine Vision Conference*.

Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., & Liu, W. (2018). Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 5265–5274).

Wang, W., Sun, Y., Zhang, W., & Yang, Y. (2021a). D^2lv: A data-driven and local-verification approach for image copy detection. arXiv preprint arXiv:2111.07090

Wang, W., Zhang, W., Sun, Y., & Yang, Y. (2021b). Bag of tricks and a strong baseline for image copy detection. arXiv preprint arXiv:2111.08004

Wang, W., Sun, Y., Yang, Y. (2023a). A benchmark and asymmetrical-similarity learning for practical image copy detection. In *AAAI Conference on Artificial Intelligence*.

Wang, W., Zhong, Z., Wang, W., Chen, X., Ling, C., Wang, B., & Sebe, N. (2023b). Dynamically instance-guided adaptation: A backward-free approach for test-time domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 24090–24099).

Wang, Z., Gao, Z., Guo, K., Yang, Y., Wang, X., & Shen, H. T. (2023c). Multilateral semantic relations modeling for image text retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2830–2839).

Wu, W., Sun, Z., Song, Y., Wang, J., & Ouyang, W. (2024). Transferring vision-language models for visual recognition: A classifier

perspective. *International Journal of Computer Vision, 132*(2), 392–409.

Zhong, Z., Zhao, Y., Lee, G. H., & Sebe, N. (2022). Adversarial style augmentation for domain generalized urban-scene segmentation. *Advances in Neural Information Processing Systems, 35*, 338–350.

Zhou, K., Yang, Y., Qiao, Y., & Xiang, T. (2023). Mixstyle neural networks for domain generalization and adaptation. *International Journal of Computer Vision*, 1–15.

Zhu, J., Liu, L., Zhan, Y., Zhu, X., Zeng, H., & Tao, D. (2023). Attribute-image person re-identification via modal-consistent metric learning. *International Journal of Computer Vision, 131*(11), 2959-2976.