




Continual Face Forgery Detection via Historical Distribution Preserving

Ke Sun¹ · Shen Chen² · Taiping Yao² · Xiaoshuai Sun¹ · Shouhong Ding² · Rongrong Ji¹ 

Received: 12 September 2023 / Accepted: 13 June 2024 / Published online: 4 September 2024

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024, corrected publication 2024

Abstract

Face forgery techniques have advanced rapidly and pose serious security threats. Existing face forgery detection methods try to learn generalizable features, but they still fall short of practical application. Additionally, finetuning these methods on historical training data is resource-intensive in terms of time and storage. In this paper, we focus on a novel and challenging problem: Continual Face Forgery Detection (CFFD), which aims to efficiently learn from new forgery attacks without forgetting previous ones. Specifically, we propose a Historical Distribution Preserving (HDP) framework that reserves and preserves the distributions of historical faces. To achieve this, we use universal adversarial perturbation (UAP) to simulate historical forgery distribution, and knowledge distillation to maintain the distribution variation of real faces across different models. We also construct a new benchmark for CFFD with three evaluation protocols. Our extensive experiments on the benchmarks show that our method outperforms the state-of-the-art competitors. Our code is available at <https://github.com/skJack/HDP>.

Keywords Face forgery detection · Continual learning · Universal adversarial perturbation

1 Introduction

Over the past decades, face forgery methods have made significant strides, capturing the interest of both the academic and industrial realms (Thies et al., 2015; Rossler et al., 2019; Dolhansky et al., 2020). These techniques have the prowess to create ultra-realistic forged faces, so convincing at times that they can easily deceive the human eye. This verisimilitude has far-reaching implications, giving rise to potential malicious misuse. Whether used in privacy infringements, identity fraud, or other deceptive practices, they pose severe societal challenges. Therefore, the need to engineer potent methods capable of differentiating real faces from their forged counterparts has become paramount.

Recently, many significant face forgery detection methods have been proposed to mine the subtle artifacts (Chen et al., 2021; Qian et al., 2020; Dolhansky et al., 2020; Dang

et al., 2020; Afchar et al., 2018; Sun et al., 2021; Luo et al., 2023, 2021) and achieved extraordinary performance under known forgery types. However, they all suffer from significant performance degradation when testing under new forgery attacks. Some methods have attempted to learn generalized representation (Li et al., 2020; Sun et al., 2022; Luo et al., 2021; Sun et al., 2021, 2023), but their performance on unknown attacks is still far from practical application. As face forgery techniques continue to evolve, acquiring a comprehensive dataset of all forgery methods and their respective manipulation techniques becomes challenging, especially since such data is often accrued over time. Given this scenario, continually updating models to integrate both past and current data can be both time-consuming and storage-intensive. Relying solely on recent forgery data for model updates introduces the risk of catastrophic forgetting—a situation where the model forgets its previously learned patterns. These limitations present significant barriers to the practical deployment of face forgery detection systems in real-world scenarios.

To address this challenge, we introduce and tackle a novel and pressing problem: **Continual Face Forgery Detection** (CFFD) (Li et al., 2023). CFFD evaluates the capacity of detectors to adapt to new attack techniques sequentially while retaining proficiency in recognizing earlier ones. As depicted in Fig. 1, diverse training data are introduced across different stages. The goal of the model is to assimilate

Communicated by Segio Escalera.

✉ Rongrong Ji
rrji@xmu.edu.cn

Ke Sun
skjack@stu.xmu.edu.cn

¹ Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, Xiamen, Fujian, China

² YouTu Lab, Tencent, China

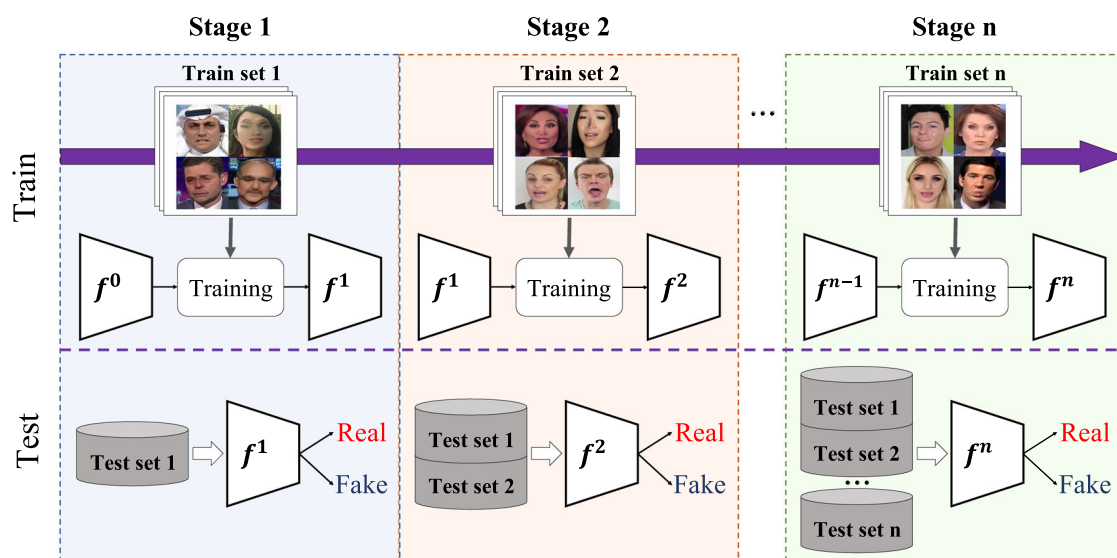


Fig. 1 Pipeline of the training and testing processes in the Continual Face Forgery Detection (CFFD) setting (Color figure online)

new information without discarding what it has previously learned. While certain existing continual learning methods might seem applicable to CFFD—such as regularization techniques (Li & Hoiem, 2017; Hou et al., 2019; Kirkpatrick et al., 2017), parameter-isolation strategies (Rusu et al., 2016; Xu & Zhu, 2018; Verma et al., 2021), and replay methods (Rebuffi et al., 2017; Aljundi et al., 2019; Chaudhry et al., 2020)—these general approaches have not been expressly tailored for face forgery detection. This oversight often culminates in a performance that is less than ideal. To remedy this, we identify two distinct characteristics that differentiate CFFD from conventional continual learning: (1). Face forgery detection is inherently a binary classification task. It is mainly concerned with differentiating an array of fake faces, implying that features of real faces tend to exhibit more uniformity. (2). The differences between real and forged faces are subtle, putting the onus on the detection network to identify the discriminative differences effectively.

In this paper, we propose a novel Historical Distribution Preserving (HDP) framework that preserves the historical distributions of real and forged faces. The main challenge is *how to efficiently preserve historical forgery distribution and the discriminative differences from real faces?* Motivated by the universal adversarial perturbation (UAP), which can be regarded as a *feature* that captures the main direction of image-space that affects the model decision (Jetley et al., 2018; Zhang et al., 2020), we propose a new perspective of continual learning that uses the UAP as the discriminative feature of forged faces relative to real faces. Specifically, instead of storing redundant forged data, we only store a single UAP generated by the historical model. And when dealing with new forgery attacks, we combine the real faces

and the UAP as the pseudo-forged faces to simulate historical forgery distribution. Such faces can not only reserve the discriminative feature of the forgery but also protect privacy from being violated. Feature-based knowledge distillation is further employed to maintain the distribution of real and pseudo-forged faces across different models to reduce the domain gap between each stage. The inherent uniformity of real-face features allows for easier maintenance of the real distribution and further ensures the restoration of historical distribution by the pseudo-forged samples.

Notably, our methodology demands minimal storage overhead and is seamlessly integrable with any classification-centric face forgery detection framework within the proposed CFFD paradigm. This bears significant ramifications for real-world implementations, where efficiency and adaptability are paramount. Extensive experiments on the CFFD benchmark show that our method not only outperforms state-of-the-art competitors but also exhibits exceptional resilience against continually evolving forgery techniques. In summary, our main contributions are as follows:

- We first exploit the Universal Adversarial Perturbation (UAP) into continual learning as the discriminative historical feature instead of storing samples.
- We propose a novel Historical Distribution Preserving (HDP) framework in the Continual Face Forgery Detection task to preserve the distributions of real and forgery faces and their discriminative distribution difference, thus mitigating catastrophic forgetting.
- We provide a benchmark with three evaluation protocols for the CFFD task. Extensive experiments and visualizations demonstrate the effectiveness of our method.

2 Related Work

2.1 Face Forgery Detection

Face forgery detection is mainly to identify whether the input face is forged or not. Early studies use hand-crafted features to seek the artifacts in forged faces (Matern et al., 2019; Yang et al., 2019). Subsequently, deep learning-based works achieve better performance by extracting high-level features for classification. For example, some works (Stehouwer et al., 2019; Zhao et al., 2021a) highlight the manipulated regions through attention mechanism, while others (Chen et al., 2021; Zhao et al., 2021b) leverage the self-consistency of local regions as forged clues. Although these methods achieve extraordinary performance on seen forgery attacks, they suffer from significant performance degradation when testing under new forgery attacks. Subsequently, some methods have attempted to learn generalized representation. Face X-ray (Li et al., 2020) conducts self-supervised learning driven by simulation of the blending traces. LTW (Sun et al., 2021) uses meta-learning to reweight samples and provide gradient regularization. DCL (Sun et al., 2022) controls the intra-class variance to preserve the transferability via contrastive learning. However, their performance on unseen attacks is still far from practical application. With the continuous emergence of new forgery methods, it is difficult to obtain all forgery samples with various manipulation techniques at once. And performing finetune directly based on historical training data often requires much time and storage costs. To address these issues, We focus on a new yet practical face forgery detection task, named Continual Face Forgery Detection (CFFD) (Li et al., 2023), which enables efficient learning to deal with continuous new attacks.

2.2 Continual Learning

Current continual learning methods can be taxonomized into three major categories (Delange et al., 2021). (1) *Parameter isolation* methods intend to assign separate components for each task. These works typically require a task oracle (Delange et al., 2021) that matches the corresponding dynamic layer to a special task, which increases the network complexity and is not conducive to practical deployment. (2) *Regularization-based* methods introduce an extra regularization term in the loss function to consolidate previous knowledge while training new tasks. Specifically, LwF uses the output of previous models as soft-label and treats the regularization term as knowledge distillation (Li & Hoiem, 2017). Other regularization-based methods estimate a distribution over the model parameters such as EWC (Kirkpatrick et al., 2017) and MAS (Aljundi et al., 2018). Though these approaches alleviate catastrophic forgetting to some extent, they may yield suboptimal performance when faced with

challenging settings or complex datasets (Wang et al., 2021). (3) *Replay-based* methods preserve the samples of previous tasks in a memory bank and replay them when learning the current task. The most classic method is iCaRL (Rebuffi et al., 2017), which stores a subset of exemplars based on the distance of the class means center in the feature space. Recently, some works (Buzzega et al., 2020; Chaudhry et al., 2020; Wu et al., 2019) combine the knowledge distillation regularization with a memory bank to further avoid overfitting. Although replay methods can alleviate catastrophic forgetting well, the performance is vulnerable to the size of buffers, and the extra memory bank may also bring data privacy leakage and storage burden. Differently, we propose a new perspective of continual learning that introduces the universal adversarial perturbation (UAP) to simulate historical distribution. Remarkably we store only one UAP generated by the historical model without any additional rehearsal buffers, while still achieving better performance than traditional replay methods.

2.3 Universal Adversarial Perturbation

Universal Adversarial Perturbation (UAP) is a unique type of noise designed to deceive deep learning models with a high success rate (Moosavi-Dezfooli et al., 2017). Unlike per-instance adversarial examples (Szegedy et al., 2013), which require different perturbations for each instance, UAP employs a single perturbation vector to mislead models across various images. (Moosavi-Dezfooli et al., 2016) first introduced and created UAP using iterative deepfool methods. The interpretability of UAP has garnered significant interest within the computer vision community. (Moosavi-Dezfooli et al., 2016) further provided a theoretical proof highlighting the presence of a low-dimensional subspace that captures the correlation with the decision boundary of the target model, elucidating the effectiveness of UAP (Chaubey et al., 2020) (Moosavi-Dezfooli et al., 2018). (Jetley et al., 2018) discussed the consistency between model performance and robustness, while (Zhang et al., 2020) illustrated that UAPs possess features that are independent of the images they aim to compromise. Building on these insights, our approach leverages UAP and relevant data to approximate prior feature distributions and incorporate them during new task training.

3 Method

3.1 Problem Definition

The goal of Continual Face Forgery Detection (CFFD) is to develop a unified detector from a sequence of face data encompassing different forgery types and domains. Formally,

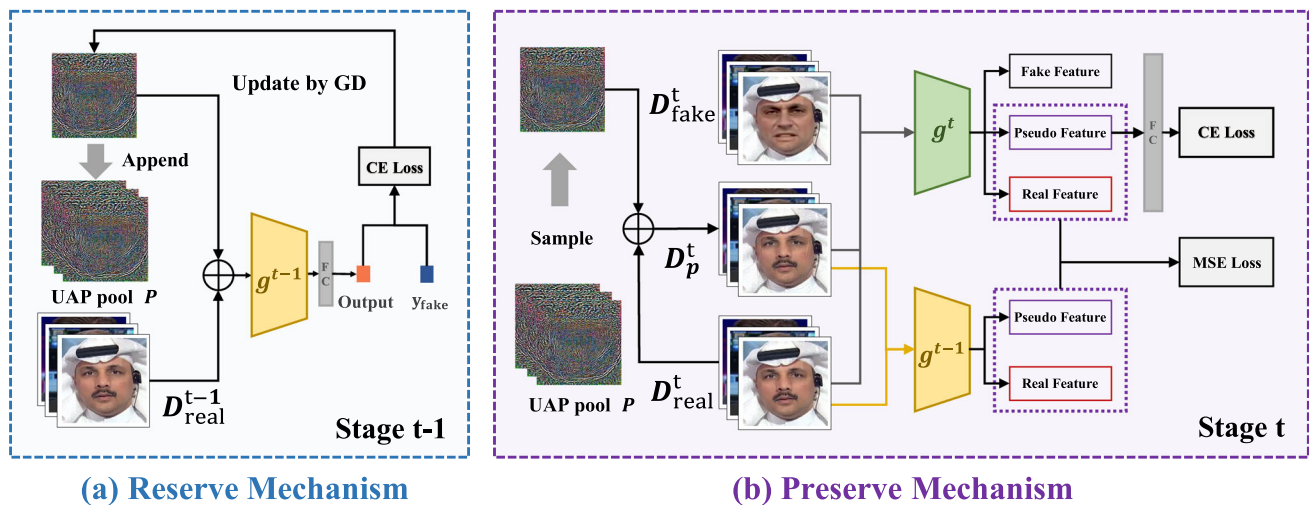


Fig. 2 Overview of the proposed Historical Distribution Preserving framework for CFFD task. Subfigure **a** elucidates the reserve mechanism and subfigure **b** details the preserve mechanism

we denote the training data available at the t -th stage as $D^t = \{D^t_{\text{real}}, D^t_{\text{fake}}\}$, where D^t_{real} and D^t_{fake} represent the incoming real and forged faces, respectively. The face forgery detection model for the t -th stage, f^t , is trained exclusively from the dataset D^t . Data from the previous stages, $\{D^i\}_{i=1}^{t-1}$, are no longer accessible. This framework is driven by two main factors: firstly, it eliminates the need for storing and processing a large volume of historical data, addressing both storage and computational issues; secondly, it considers the privacy concerns associated with using historical facial data. For evaluation, the model f^t is tested against all previously identified forgery attacks, providing a thorough examination of its effectiveness and flexibility in a wide array of situations.

3.2 Base Training

In this section, we detail the fundamental strategy for training the face forgery detection model. Practically, our proposed method is designed to be seamlessly integrated into any existing state-of-the-art (SOTA) classification-based technique. For the sake of clarity, we utilize the well-established binary classifier model (Dolhansky et al., 2020), optimizing it on dataset D using cross-entropy loss, which is expressed as:

$$L_{ce} = -\frac{1}{|D|} \sum_{(x,y) \in D} y \log(f(x)) + (1-y) \log(1-f(x)), \quad (1)$$

where $x \in \mathbb{R}^{C \times H \times W}$ and $y \in \{0, 1\}$ denote the images and their corresponding labels (real/fake) from the dataset D , respectively. At the t -th stage, if we employ cross-entropy loss directly on $D^t = \{D^t_{\text{real}}, D^t_{\text{fake}}\}$, the model f^t will recognize the forgery types it has been exposed to but will likely

forget about earlier forgery techniques. Therefore, the critical challenge in CFFD is to maintain the historical forgery distribution and the distinct differences between real and forged faces.

3.3 Overall Framework

To address the issue of catastrophic forgetting in continual face forgery detection, we present the Historical Distribution Preserving (HDP) framework, comprising two principal mechanisms: the Reserve Mechanism and the Preserve Mechanism, as illustrated in Fig. 2. The Reserve Mechanism employs Universal Adversarial Perturbation (UAP) (Moosavi-Dezfooli et al., 2017) to capture the distinct features that differentiate forged faces from real ones. We generate UAP using the model f^{t-1} , which has been effectively trained up to the $t-1$ stage, capturing the essential features of forgery attacks encountered so far and storing them in a UAP pool.

Transitioning to a new attack at the t stage, we deploy the Preserve Mechanism. This strategy begins by melding real faces with the generated UAPs, simulating the distribution of forgery attacks from previous stages. Subsequently, this synthesized data is trained in tandem with the fresh dataset. Furthermore, we conduct feature-wise knowledge distillation between pseudo and real features across the current and the last step model to maintain the stability of the distribution in all stages.

3.4 Reserve Mechanism

The main goal of the reserve mechanism is to maintain the historical distribution of forgery techniques by blending Universal Adversarial Perturbation (UAP) with real data. Prior

research (Zhang et al., 2020; Jetley et al., 2018) has shown that UAP represents a unique attribute that introduces directional perturbations across class boundaries, independent of varied individual features. Jetley et al. (2018) further clarified the strong link between certain directions and class identities, indicating that the network perceives these directions as intrinsically linked to class identities. Therefore, we use UAP as a key feature to distinguish between forged and real faces, merging it with real data to create pseudo-forged examples. For a model f^{t-1} trained during the $t-1$ phase, which incorporates a feature extractor g^{t-1} , an FC layer, and real subset samples $x_{\text{real}}^{t-1} \in D_{\text{real}}^{t-1}$ labeled $y_{\text{real}} = 0$, our goal is to ascertain perturbation vectors $p^{t-1} \in \mathbb{R}^{C \times H \times W}$ capable of misleading the model into misclassifying real images as fake. Analogous to Moosavi-Dezfooli et al. (2017), we target a vector p^{t-1} that conforms to:

$$\begin{aligned} \text{Pred}(f^{t-1}(x_{\text{real}}^{t-1} + p^{t-1})) &\neq y_{\text{real}} \\ \text{s.t. } \|p^{t-1}\|_{\infty} &\leq \epsilon, \end{aligned} \quad (2)$$

where Pred converts the logit into a prediction value (i.e., 0 for real and 1 for fake) and ϵ adjusts the magnitude of the perturbation vector. The pseudo-forged samples x_p^{t-1} are represented as:

$$x_p^{t-1} = x_{\text{real}}^{t-1} + p^{t-1}. \quad (3)$$

Unlike the original UAP method (Moosavi-Dezfooli et al., 2017), which uses deepfool (Moosavi-Dezfooli et al., 2016) to identify perturbations that delineate the decision boundary, we view UAP as a powerful feature. Taking a cue from Zhang et al. (2020), we utilize the gradient descent technique to refine p^{t-1} by minimizing the entropy between pseudo-forged samples x_p^{t-1} and the fake label $y_{\text{fake}} = 1$. This process is encapsulated as:

$$p^{t-1} = p^{t-1} - \alpha * \text{sgn}(\nabla_{p^{t-1}} \log(f^{t-1}(x_p^{t-1}))), \quad (4)$$

where sgn represents the symbolic function, and α signifies the learning rate of p^{t-1} . The perturbation is reiterated until x_p^{t-1} transgresses the decision boundary. The algorithm concludes when the number of successfully modified prediction outcomes surpasses the predetermined threshold σ . After obtaining the UAP p^{t-1} , we integrate it into the UAP pool P in preparation for the preserve phase.

3.5 Preserve Mechanism

After reconstructing the historical distribution, the next step focuses on maintaining the forgery distribution by incorporating it into the current training cycle. Specifically, in the t -th training stage, we apply entropy optimization to the pseudo-forged samples along with their assigned fake labels. This

process incorporates a regularization term, which guarantees the distribution of the pseudo-forged samples is effectively preserved.

While training the model f^t using the dataset D^t , UAPs from previous stages up to $t-1$, denoted as $\{p^n\}_{n=1}^{t-1}$, are randomly sampled from the UAP pool P at the start of each training iteration. Subsequently, a pseudo-forged dataset D_p^n is created by adding p^n to D_{real}^t on a pixel-wise basis, formulated as $D_p^n = p^n + D_{\text{real}}^t$, and assigned the label $y_{\text{fake}} = 1$. The entropy between D_p^n and the fake label y_{fake} is calculated as follows:

$$E^n = -\frac{1}{|D_p^n|} \sum_{x_p^n \in D_p^n} \log(f^t(x_p^n)). \quad (5)$$

Algorithm 1 Historical Distribution Preserving

Input: Sequence of training dataset $D = \{D_{\text{real}}, D_{\text{fake}}\}_{t=1}^T$, initial model f^1 , UAP pool P

Init: $P \leftarrow \{\}$

- 1: **for** $\{D_{\text{real}}, D_{\text{fake}}\} \in D$ **do**
- 2: **if** $t == 1$ **then**
- 3: Optimize model f^t with Eq. 1 on D^t to obtain model f^{t+1} .
- 4: **else**
- 5: Optimize model f^t with Eq. 8 on D^t to obtain model f^{t+1} .
- 6: **end if**
- 7: Generate UAP p^{t+1} for model f^{t+1} .
- 8: $P \leftarrow p^{t+1}$
- 9: **end for**
- 10: **Return** model f^{T+1}

To calculate the total training entropy E^t at stage t , it's essential to account for the collective impact of the entropies E^n randomly sampled from the UAP pool. Therefore, the overall training entropy reflects not only the current stage's influence but also the aggregated impact of previous stages.

This ensures that the distribution of forgery attacks from the previous stage is perpetuated. Nevertheless, given that parameters undergo dynamic optimization, the consistency of the pseudo-forged distribution might be compromised. Addressing this concern, we implement feature-wise knowledge distillation, utilizing the prior feature extractor g^{t-1} . This maintains distillation based on the pseudo-forged samples x_p^t , and is formalized as:

$$L_p^t = \frac{1}{|D_p^t|} \sum_{x_p \in D_p^t} \|g^t(x_p) - g^{t-1}(x_p)\|^2. \quad (6)$$

In face forgery detection task, the focus is primarily on identifying signs of forgery instead of distinguishing between real and fake facial content. In real-world scenarios, authentic images tend to be more uniform than their forged counterparts. However, emerging forgery methods can shift the

distribution of real images, affecting the creation of pseudo-forged samples. To mitigate this issue, it's crucial to ensure the consistency of real feature distillation. Following the same paradigm as L_p , a single-set feature-based knowledge distillation is applied. Without any compromise to generality, let g^t be the feature extractor for the model f^t . Here, the previous feature extractor, g^{t-1} , is regarded as the teacher, utilizing the mean-squared function as the regularization term. This can be articulated as:

$$L_r^t = \sum_{x \in D_{real}^t} \|g^t(x) - g^{t-1}(x)\|^2. \quad (7)$$

By applying regularization at each training stage with the model from the previous phase, we ensure the distribution of real samples remains stable and consistent.

3.6 Loss Function

In summary, when new forgery attacks arise at the t stage, the overall loss function for our proposed method of preserving historical distribution is defined as the sum of the base training loss and the aforementioned distribution-preserving optimizations. This can be expressed as:

$$L^t = L_{ce}^t + E^t + \beta(L_r^t + L_p^t), \quad (8)$$

where β is a hyper-parameter that weighs the regularization terms. To offer a clearer understanding of our method, we provide the pseudo-code of the HDP in Algorithm 1.

4 Experiment

4.1 Experimental Setting

Datasets and Data Availability Statement We conduct experiments based on four challenging datasets: FaceForensics++ (Rossler et al., 2019) (FFpp) is a widely-used dataset containing four different face synthesis methods, including two deep learning-based methods Deepfakes (DF) and NeuralTextures (NT) and two graphics-based approaches Face2Face (F2F) and FaceSwap (FS). Celeb-DF (Li et al., 2020) is another Deepfake dataset that collects real source videos from Youtube and generates forgery videos using an improved DeepFake synthesis method, resulting in a higher quality of forgery samples. DFDC (Dolhansky et al., 2020) is a large-scale dataset with various manipulated methods and backgrounds. Wild-Deepfake (Zi et al., 2020) is a face forgery dataset where all videos are obtained from the internet, thus it has various synthesis methods, identities, and image qualities.

CFFD Benchmark We've introduced the CFFD benchmark, which comprises three distinct protocols, each progressively more challenging than the last. Below, we outline each protocol in detail:

Protocol 1 This protocol comprises four fake subsets from FFpp. Though these subsets maintain consistency in context and individual identity, they vary in their manipulation types. The core objective of Protocol 1 is to emulate the continuous emergence of novel attack strategies.

Protocol 2 Diversifying the range, this protocol draws from four fake datasets, namely FFpp, Celeb-DF, Wild-Deepfake, and DFDC. When juxtaposed with Protocol 1, Protocol 2 exhibits a wider array of attack types and more pronounced domain gaps. The real samples in this protocol differ at each stage, which elevates its difficulty. This intricacy, combined with its expansive applicability, positions Protocol 2 as particularly challenging.

Protocol 3 Tailored to assess the ability to handle extended sequences of evolving real and fake imagery, this protocol unfolds across ten stages. It utilizes subsets from FFpp and evenly apportions the datasets from Celeb-DF, Wild-Deepfake, and DFDC. This layout accentuates the risk of catastrophic forgetting. Between the three protocols, Protocol 3 most accurately mirrors the complexities of real-world scenarios, typified by extended sequences and multi-domain data. For an in-depth exploration of the three protocols, please refer to the supplementary materials.

Evaluation Metrics Our primary evaluation metrics encompass the accuracy score (ACC) and the area under the receiver operating characteristic curve (AUC). Both ACC and AUC are standard measures to evaluate the performance of classification models. While ACC measures the proportion of correctly classified instances out of the total instances, AUC provides an aggregate measure of the model's performance across all possible classification thresholds.

In the context of the CFFD benchmark, we employ two primary metrics:

AVG (Average Performance) represents the performance of the *last model* over each individual task. By evaluating the last model across the various tasks, we aim to understand how well the continually updated model performs on different challenges, especially the most recent ones. In mathematical terms, AVG calculates the mean of the performance metric (be it ACC or AUC) of the last model on each separate task.

PRE (Previous Performance) is designed to gauge the model's resilience against catastrophic forgetting. Catastrophic forgetting refers to the model's tendency to overwrite

previous knowledge when learning new information. PRE, therefore, calculates the average performance metric of the model on all previous tasks, relative to the current task. A high PRE score indicates that the model retains its proficiency on prior tasks even after being updated with new tasks, showcasing its resistance to catastrophic forgetting.

Implementation Details To ensure a fair comparison, all comparative methods are aligned to the following settings, unless specified otherwise. We utilize the EfficientNet-b4 (Tan & Le, 2019), pre-trained on ImageNet (Deng et al., 2009), as the backbone for our face forgery detection model. DSFD (Li et al., 2019) serves as our face detector across all datasets. All input face images are resized to a resolution of 224×224 . For model training, we employ the Adam optimizer. Key parameters are set as follows: weight decay is $1e - 5$, the learning rate is 0.0001, and the batch size is fixed at 64. While generating the UAP, we adjust the norm of perturbation (ϵ) to 0.15, set α at 0.0001, and determine the successful threshold (σ) to be 0.8. The complete framework is realized using PyTorch and is run on a single NVIDIA A-100 GPU.

4.2 Quantitative Results

Comparing Methods We benchmark our method against several state-of-the-art continual learning techniques: including (1) Sequence fine-tune (SFT): Directly fine-tune model on new task dataset without extra operation; (2) Learning without forgetting (LwF) (Li & Hoiem, 2017): A regularization-based technique that retains previous knowledge via knowledge distillation. (3) Memory Aware Synapses (MAS) (Aljundi et al., 2018): Another regularization-based strategy that forms the regularization term using the previous model parameters and their corresponding importance weight. (4) Nearest Class Mean Classifier for real sample ((NCM*)) (Zhang et al., 2020): This method adjusts the unified classifier to our setting, constraining the real cluster centers and using the NCM classifier in place of the softmax-fc layer. (5) iCaRL (Rebuffi et al., 2017): A well-known replay method that chooses to best approximate class means in the feature space and stores them in a memory bank. (6) Supervised Contrastive Replay (SCR) (Mai et al., 2021): A recent replay technique that merges supervised contrastive learning with the NCM classifier. (7) Continual Representation using Distillation (FReTAL) (Kim et al., 2021): The first approach that considers continual learning settings within the face forgery detection task. (8) Dynamically Expandable Representation (DER) and DER++ (Yan et al., 2021): A unique two-stage learning approach that employs dynamically expandable representation for more effective incremental concept modeling. (9) Contrastive Continual Learning (Co2L) (Cha et al., 2021): This method learns

representations through the contrastive learning objective and preserves these representations using a self-supervised distillation step. (10) Feature Boosting and Compression method (Foster) (Wang et al., 2022): A simple yet effective approach that gradually fits the residuals between the target model and the previous ensemble model.

(11) Prototype Reminiscence and Knowledge Aggregation (PRKA) (Shi & Ye, 2023): A prototype reminiscence mechanism that incorporates the previous class prototypes with data augmentation. (12) Prediction Error-based Classification (PEC) (Zajac et al., 2024): This method allows adaptive coefficients during training, thereby always achieving the tightest bound.

For all the replay methods mentioned above, we set the memory buffer size to hold 500 forgery images for each task.

Quantitative Results on Protocol1 Since sequence order is agnostic in continual learning, two orders are evaluated in protocol 1. *Order1* is represented by $\mathbf{DF} \rightarrow \mathbf{F2F} \rightarrow \mathbf{FS} \rightarrow \mathbf{NT}$ and *Order2* is $\mathbf{NT} \rightarrow \mathbf{DF} \rightarrow \mathbf{F2F} \rightarrow \mathbf{FS}$. The adjacent attack types in *Order1* are more alike, in contrast to *Order2*, where the manipulation methods between stages are distinct. The results of *Order1* are shown in Tab. 1, while those of *Order2* are presented in the Appendix. In In Tab. 1, it's evident that our proposed approach significantly surpasses the competing methods in terms of both ACC and AUC metrics. Relative to the SFT, which grapples with pronounced catastrophic forgetting, our method sees approximately a 27% enhancement in the PRE metric. This uplift primarily stems from the introduced historical distribution preservation strategy. Further, the performance of regularization-centric strategies, like LwF and MAS, is hampered by the pronounced domain gap between FaceSwap and NeuralTextures - a challenge our method can adeptly navigate. Unlike existing replay strategies that demand large buffer sizes, our method requires storing only one UAP per task, giving us an advantage in preventing forgetting with minimal storage overhead. Furthermore, our approach surpasses the latest PEC method by 11% on the PRE metric, while PEC employs multiple student models. We utilize a single student model distillation, demonstrating the efficacy of our method. Simultaneously, our method outperforms the recent memory-based representative method, Foster, by a margin of 5% points in AVG in terms of ACC without using a buffer. This further underscores the superiority of UAP in restoring historical distributions. To further show the ability to combat forgetting, we illustrate the average performance of the previous task and current task during each training process on *order1* and *order2*, respectively. As shown in Fig. 3a, b, our methods can achieve state-of-the-art results under different stages on both ACC and AUC. Figure 3c, d summarizes the results on the *order2*. Similar to the *order1*, our method achieves the SOTA performance of AVG and PER compared with other methods on

Table 1 Quantitative results on protocol1 in terms of ACC and AUC

Method	Buffer/step	Deepfakes		Face2Face		FaceSwap		NeuralTextures		AVG		PRE	
		ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
SFT	0	76.88	91.88	68.15	85.57	59.74	72.78	<u>95.53</u>	98.43	75.07	87.16	67.29	88.36
LwF	0	81.51	95.61	72.26	91.81	61.27	81.38	95.48	<u>98.28</u>	77.63	91.77	74.09	94.05
MAS	0	86.97	95.14	78.01	86.81	65.89	72.89	90.86	96.15	80.43	87.74	83.76	90.75
NCM	0	76.94	92.90	67.67	92.99	61.77	70.75	95.66	96.81	75.53	85.28	69.13	87.73
PRKA	0	82.84	95.21	76.66	84.57	89.93	95.46	91.33	98.02	85.19	93.31	81.37	90.57
PEC	0	83.57	96.13	79.93	86.73	78.85	91.25	94.05	98.19	84.09	93.07	83.50	93.78
iCaRL	500	84.46	95.67	81.71	93.28	73.41	87.01	94.70	98.15	83.57	93.52	82.82	93.84
SCR	500	82.85	<u>96.75</u>	81.42	94.79	62.96	89.49	95.53	97.33	80.69	86.75	75.29	<u>95.27</u>
FReTAL	500	86.95	94.26	<u>88.55</u>	93.65	<u>88.50</u>	<u>94.85</u>	94.43	98.25	<u>89.53</u>	<u>95.25</u>	80.07	95.24
DER	500	78.41	87.82	73.32	82.63	71.80	74.24	88.91	93.14	77.86	84.45	80.50	89.10
DER++	500	81.20	94.10	74.93	88.90	70.15	76.20	90.23	97.40	79.12	88.64	80.60	89.09
Co2L	500	84.91	94.81	76.50	86.81	64.99	73.95	94.36	97.64	80.19	88.30	77.52	89.00
Foster	500	<u>87.55</u>	95.93	82.16	<u>95.01</u>	85.12	92.41	93.32	95.88	87.01	94.80	<u>84.91</u>	92.51
HDP	0	93.10	98.17	91.90	96.86	93.47	97.52	94.81	97.12	93.32	97.41	94.94	98.50
Joint	–	96.99	99.75	96.35	99.12	96.63	99.30	94.04	97.84	96.00	99.00	–	–

We evaluate the last model on all the previous test sets. The buffer represents the memory bank budget save the previous images per stage. The underlined values represent the second best result. The values in bold represent the best results

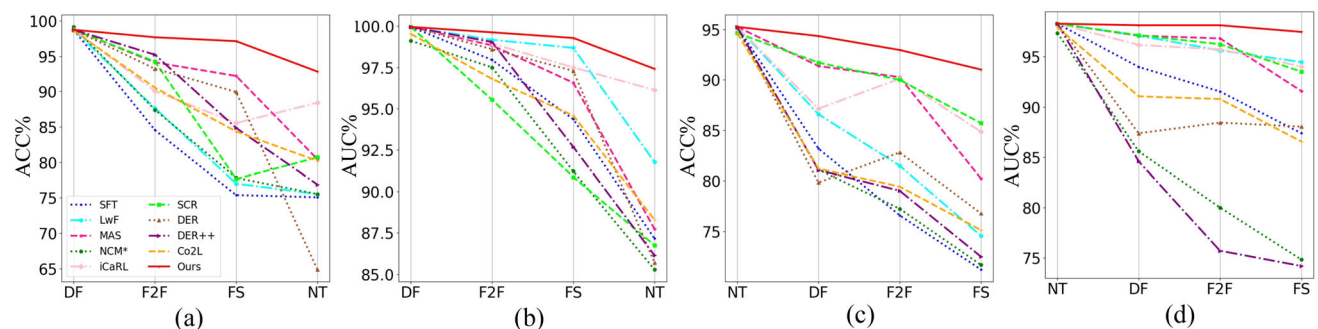


Fig. 3 Illustration of the non-forgetting evaluation of each stage on protocol 1. **a, b** show the trend of average AUC and ACC for the previous dataset during the training process for order 1, respectively. **c, d** represent the order2

order2. Specifically, when facing the more difficult order, the ACC of AVG and PRE still obtain 20% and 25% improvement compared with SFT. The detailed results of Order2 are presented in the Appendix. The above results demonstrate the effectiveness and robustness of our approach to different task sequencing.

Quantitative Results on Protocol2 In Protocol 2, which is characterized by a substantial domain gap, the input order is set as **FFpp→Celeb-DF→WildDeepfake→DFDC**. As seen in Tab. 2, our method notably surpasses other comparative methods in terms of both ACC and AUC for AVG and PRE metrics. Specifically, the ACC for PRE registers at 86.99%, marking a 7% enhancement from the baseline SFT. Our technique achieves state-of-the-art (SOTA) performance

without any buffer, distinguishing it from other methods reliant on buffers. In contrast to buffer-based strategies like FReTAL, which reserves 500 images per stage, our method attains a 3% boost in the PRE metric without retaining any historical data. Additionally, when we augment our HDP to accommodate historical data with a buffer size of 50 images per stage, our method consistently outperforms competitors, even with ten times less storage. Moreover, our approach achieves a 1.89% higher PRE metric than the Foster method, which utilizes 500 samples per round, without employing an additional buffer. Moreover, our method significantly outperforms methods based on various prototypes, such as PRKA and NCM, indicating that UAP might offer a better feature representation than traditional prototype-based features at

Table 2 Quantitative results on protocol2

Method	Buffer/step	FFpp		Celeb-DF		Wild-Deepfake		DFDC		AVG		PRE	
		ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
SFT	0	85.02	93.88	81.75	90.17	72.59	83.27	87.74	<u>95.67</u>	81.77	90.74	79.92	89.38
LwF	0	87.08	96.64	83.82	90.69	73.77	84.85	87.12	94.84	82.94	91.75	82.93	92.84
MAS	0	85.45	<u>97.16</u>	81.17	89.29	76.91	81.03	87.03	90.10	81.86	89.39	82.64	92.00
NCM	0	85.83	93.44	83.76	90.22	77.01	83.72	86.56	93.76	83.29	90.28	80.38	90.14
PRKA	0	87.11	94.05	84.49	91.85	75.54	80.19	87.49	94.22	83.65	90.07	83.27	93.09
PEC	0	88.90	95.78	85.39	92.05	76.78	81.15	87.12	94.10	84.39	90.77	84.59	93.35
iCaRL	500	85.50	95.62	85.07	92.77	74.87	84.49	87.81	95.46	83.38	92.08	84.21	93.15
SCR	500	86.90	96.46	86.01	92.99	76.71	85.59	<u>88.19</u>	95.63	84.45	92.66	83.30	93.97
FReTAL	500	<u>90.67</u>	96.78	85.62	94.26	76.58	86.34	87.12	95.52	84.99	93.22	83.70	93.43
DER++	500	87.89	94.40	84.92	91.05	73.58	85.48	87.29	95.35	83.42	91.32	83.35	93.77
Co2L	500	87.69	95.77	84.57	90.08	73.79	81.20	86.30	91.77	83.08	89.70	83.07	92.57
Foster	500	87.97	95.32	86.77	<u>94.57</u>	77.87	<u>87.89</u>	86.09	94.78	84.67	93.14	85.10	93.81
HDP	0	87.81	96.48	<u>86.91</u>	93.20	<u>78.03</u>	87.21	87.51	95.05	<u>85.06</u>	<u>92.98</u>	<u>86.99</u>	<u>94.09</u>
HDP	50	91.57	98.26	87.68	94.98	80.31	87.91	88.86	95.77	86.53	94.23	88.57	95.31
Joint	–	96.05	99.67	98.53	99.89	83.39	90.40	87.56	96.60	91.38	96.64	–	–

We evaluate the last model on the test sets of all tasks in terms of ACC and AUC. The training order is FFpp → Celeb-DF → WildDeepfake → DFDC. The buffer represents the memory bank budget save the previous images per stage. Underline indicate the sub-optimal results. The values in bold represent the best results

Table 3 Quantitative results on protocol3

Method	Buffer/stage	AVG		PRE	
		ACC	AUC	ACC	AUC
SFT	0	72.51	80.30	68.90	76.64
LwF	0	74.32	83.81	70.99	78.58
MAS	0	73.88	83.56	69.89	78.92
NCM	0	72.98	81.25	70.37	77.38
PRKA	0	74.36	84.41	71.10	77.99
PEC	0	75.08	85.22	72.47	78.53
iCaRL	500	74.17	83.50	72.77	79.94
SCR	500	75.66	85.61	72.79	80.69
FReTAL	500	76.18	84.97	73.00	81.53
DER++	500	73.87	82.21	71.19	78.98
Co2L	500	75.66	84.43	72.95	81.43
Foster	500	76.39	85.10	73.16	81.88
HDP	0	<u>77.17</u>	<u>85.94</u>	<u>74.10</u>	<u>82.01</u>
HDP	50	80.15	87.97	75.47	83.82

We evaluate the last model on the test sets of all tasks in terms of ACC and AUC. The buffer represents the memory bank budget save the previous images per stage. Underline indicate the sub-optimal results. The values in bold represent the best results

the feature level. This also demonstrates the potential of our method to be integrated with other similar approaches.

These results attest to our method’s prowess in accurately approximating the forgery distribution and retaining knowledge adeptly, even in environments with pronounced domain gaps. Importantly, our method doesn’t just alleviate catas-

trophic forgetting but also matches the performance levels of contemporary forgery attacks (from the latest stage) seen in other methodologies.

Quantitative Results on Protocol 3.

To rigorously assess our method’s performance across longer sequences, we executed comparative tests using the more complex Protocol 3. The sequence followed is **DF** → **Celeb-DF1** → **NT** → **WildDeepfake1** → **FS** → **DFDC1** → **F2F** → **Celeb-DF2** → **WildDeepfake2** → **DFDC2** with ‘1’ and ‘2’ indicating separate portions of the original dataset. Results for both AVG and PRE, expressed in terms of ACC and AUC for Protocol 3, are detailed in Tab. 3. Evidently, our method attains state-of-the-art (SOTA) results irrespective of buffer size usage. When pitted against non-buffer reliant strategies like LwF, HDP’s PRE outclasses by a 4% margin in ACC. We also tailored our HDP with a buffer size of 50 samples per stage, akin to our adaptation in Protocol 2. When juxtaposed with methods that utilize buffers of 500 samples, our strategy yields superior outcomes using just 50 samples at each stage. For instance, when gauging in terms of ACC, HDP surpasses FReTAL by 4% in AVG and 2% in PRE. Furthermore, it is observed that in cases of longer sequences, the Foster method, employing gradient boosting with a 500-sample buffer, achieved commendable results. Despite this, compared to the HDP framework that uses only a 50-sample buffer size, the AVG performance still drops by about 4%, which underscores the robustness of our method in extended sequences. In summary, these findings underline the potency

Table 4 Quantitative results on protocol 1 on three SOTA face forgery detection models in terms of ACC and AUC

Method	Deepfakes		Face2Face		FaceSwap		NeuralTextures		AVG		PRE	
	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
EN-B4	76.88	91.88	68.57	85.57	59.74	72.78	95.53	98.43	75.07	87.16	67.29	88.36
EN-B4+HDP	93.10	98.17	91.90	96.86	93.47	97.52	94.81	97.12	93.32	97.41	94.94	98.50
F3-Net	80.94	93.04	69.71	82.70	63.55	74.03	94.55	97.89	77.18	86.91	69.33	89.57
F3-Net+HDP	91.65	96.26	89.06	94.70	90.99	96.15	94.21	97.97	91.49	96.22	93.51	97.77
MAT	77.13	92.30	67.75	83.33	66.75	73.49	95.20	98.54	76.81	86.90	71.96	90.72
MAT+HDP	92.55	97.14	90.76	95.26	94.23	97.94	95.23	98.07	93.13	96.84	94.36	98.65
SIA	78.52	93.68	71.29	87.21	61.15	75.03	95.44	98.40	76.60	88.58	72.39	91.28
SIA+HDP	92.33	98.01	91.11	95.59	93.01	97.31	95.47	97.39	92.98	97.07	95.30	97.58

We evaluate the last model on all the previous test sets. The values in bold represent the best results

Table 5 Ablation study on UAP attack rate

UAP attack rate	AVG		PRE	
	ACC	AUC	ACC	AUC
0.6	90.35	96.24	90.33	96.07
0.7	91.03	96.58	92.34	97.30
0.8	93.32	97.41	94.94	98.50
0.9	92.89	97.13	94.10	97.73
1.0	92.37	97.29	93.67	97.25

The values in bold represent the best results

Table 6 Ablation study on UAP attack rate

UAP attack rate	AVG		PRE	
	ACC	AUC	ACC	AUC
0.6	90.35	96.24	90.33	96.07
0.7	91.03	96.58	92.34	97.30
0.8	93.32	97.41	94.94	98.50
0.9	92.89	97.13	94.10	97.73
1.0	92.37	97.29	93.67	97.25

The values in bold represent the best results

of our proposed HDP in long sequence environments, affirming its prospective utility in real-world applications.

Comparison with Existing Face Forgery Detection Methods. To highlight the adaptability of our HDP, we incorporated it into four cutting-edge face forgery detection methodologies: F3Net (Qian et al., 2020), Mult-Attentional (MAT) (Zhao et al., 2021a), SIA (Sun et al., 2022), UIA-VIT (Zhuang et al., 2022) and UCF (Yan et al., 2023). As shown in Tab. 4, the standalone state-of-the-art methods exhibit varying degrees of forgetting and generalization capabilities. While the latest method, UCF, achieves some generalization improvements, it still suffers from significant forgetting. However, when these methods are augmented with our HDP approach, their performance in both AVG and PRE metrics improves substantially. In particular, integrating HDP into MAT leads to a 23% improvement in the PRE metric (in terms of ACC), while F3-Net experiences an approximate 16% enhancement in AVG.

To further measure the generalization ability, we also evaluated the Unseen metric, which represents the average performance of the model on unseen attack methods during the continual learning process. By incorporating our HDP approach, the generalization performance of all methods is notably enhanced. For instance, when UCF is combined with HDP, the Unseen metric improves by 1%, indicating that

our method enhances the model's ability to handle unseen attacks. This can be attributed to our approach of using UAP for data augmentation when training on current attack data, which enhances the network's ability to learn a broader forgery distribution and capture more essential forgery features. A more detailed analysis can be found in Appendix B. These outcomes not only underscore our method's effectiveness but also its broad applicability and flexibility in improving both the forgetting resistance and generalization capability of various state-of-the-art face forgery detection methods.

4.3 Ablation Study

Ablation study on components. The optimization process of our HDP considers three main equations: Eq. 5 which targets the training of pseudo-forged samples, Eq. 6 focusing on the distillation of pseudo-forged features, and Eq. 7 for real feature distillation. An ablation study was conducted to understand the contributions of each of these equations to the model's performance. The AVG and PRE metrics under Protocol1 in terms of ACC and AUC are shown in Tab. 5. Our observations are multifold: (1). When the regularization of the real distribution is absent, the model witnesses a marked performance decline. This can be attributed to the

Table 7 Ablation study on UAP generation method

UAP method	AVG		PRE	
	ACC	AUC	ACC	AUC
Deepfool	91.31	96.72	91.64	97.88
PD-UA	93.29	96.83	92.75	97.97
Ours	93.32	97.41	94.94	98.50

The values in bold represent the best results

distribution of pseudo-forged samples, which tends to sway in tandem with shifts in the genuine distribution. This is evident upon contrasting the second, third, and fifth rows. (2). An analysis of the fourth row suggests that a constrained focus on real sample distillation yields sub-optimal results. Consequently, the model's proficiency to accurately discern forgery distributions is compromised. (3). Notably, the model's pinnacle of efficacy is attained when all three equations are concurrently and cohesively activated. This substantiates the inherent interdependence of these equations, underscoring the superior results borne from their synergy.

Ablation study on attack rate. the generated UAP can represent the discriminative feature of the training set, but there exist many out-of-distribution samples. Thus, the UAP Attack rate, which represents the proportion of successful attacks when generating UAP, is important in our method. To investigate the best attack rate, we conduct an ablation study on it and reported in Tab. 6. We varied the attack success rate within [0.6, 0.7, 0.8, 0.9, 1.0], the best results were obtained when the attack rate is 0.8. Since there are some outliers in the training dataset, the attack rate cannot be set too high, otherwise it will cause overfitting. If the attack rate is lower than 0.8, the ability of UAP to represent the distribution will decrease, which will affect the simulation of the distribution and cause knowledge forgetting.

Ablation study on UAP. To further examine the impact of different UAP generation techniques on our results, we selected two distinct UAP generation methods for comparison, replacing our original approach: Deepfool (Moosavi-Dezfooli et al., 2017) and PD-UA (Liu et al., 2019). Deepfool optimizes adversarial noise using a hyperplane-based classification method and represents the most basic version and PD-UA employs a method that utilizes priors and uncertainty estimates to generate perturbations. We can observe from Tab. 7 that the gradient-based method outperforms the traditional Deepfool approach. Additionally, compared to the more advanced PD-UA method, our simple gradient-based optimization method was 2% higher on the PRE metric. This may be due to the circular prior being more suitable for natural images than for face images. Overall, compared to the other components in Tab. 5, the choice of the UAP generation method does not significantly fluctuate the results.

Table 8 Ablation study on distillation method

Distillation	AVG		PRE	
	ACC	AUC	ACC	AUC
KL-Opt	87.10	95.79	87.52	93.36
MSE-Opt	88.17	96.79	89.21	94.17
Sim-Feat	90.05	93.81	91.77	95.69
MSE-Feat	93.32	97.41	94.94	98.50

*-Feat represents the feature distillation, *-Opt represents the distillation from model output. The values in bold represent the best results

Table 9 Ablation study on sample strategy

Sampling	AVG		PRE	
	ACC	AUC	ACC	AUC
Turn-Epoch	85.37	92.99	83.23	90.96
Turn-Iter	91.48	96.45	91.89	97.70
Random-Epoch	87.55	93.28	86.47	92.33
Random-Iter	93.32	97.41	94.94	98.50

*-Iter represents the sample per iteration, *-Epoch represents the sample per epoch. The values in bold represent the best results

Ablation study on distillation method. To investigate the impact of different distillation approaches, we compared three methods: (1) KL-Opt: the most conventional distillation method, which uses KL divergence to constrain the outputs of the student and teacher models; (2) MSE-Opt: which performs MSE regression on the outputs of the student and teacher models; (3) Sim-Feat: follows (Tung & Mori, 2019), constraining the self-similarity of features between teacher and student; (4) MSE-Feat: directly applies MSE constraints on the features of student and teacher. The final results, as shown in Tab. 8, reveal that methods imposing constraints at the feature level outperform those that apply constraints at the output level. This may stem from our distillation aim to preserve real face feature distribution, diverging from traditional distillation's focus on dark knowledge. Moreover, we found that directly using the MSE method yields better performance compared to regression based on feature self-similarity. In summary, this experiment demonstrates that feature-level distillation offers the most significant benefits for our approach.

Ablation study on sampling strategies. In the HDP approach, a UAP is randomly sampled from the UAP pool at each iteration to perform the Preserve Mechanism. To ablation the impact of various sampling strategy, Tab. 9 compares several sampling strategies: (1) Turn-Epoch: samples UAPs from the pool in turns on an epoch basis; (2) Turn-Iter: samples on an iteration basis; (3) Random-Epoch: randomly samples from the pool on an epoch basis; (4) Random-Iter: randomly samples on an iteration basis. The quantitative results suggest that

Table 10 Analysis of time consumption of each method

Method	Buffer	Epoch	Total Time	AVG	PRE
SFT	0	20	2500 s	75.07	67.29
LwF	0	20	2900 s	77.63	74.09
MAS	0	20	3080 s	80.43	83.76
NCM	0	20	3140 s	75.53	69.13
iCaRL	500	20	3517 s	83.57	82.82
SCR	500	20	3497 s	80.69	75.29
FReTAL	500	20	4437 s	89.53	80.07
DER++	500	20	4357 s	79.12	80.80
Co2L	500	20	4557 s	80.19	77.52
HDP	0	20	4096 s	93.32	94.94
HDP	0	10	2136 s	91.32	90.63

Total time represents time spent at each stage. The buffer represents the memory bank budget saved for the previous images per stage. We use ACC as the metric of AVG and PRE. The values in bold represent the best results

sampling on an iteration basis outperforms epoch-based sampling, possibly because epoch-based sampling might lead the network to overfit to a specific historical distribution, while finer-grained adjustments prevent excessive fitting. Additionally, random sampling methods surpass sequential sampling in effectiveness. This improvement is likely due to random sampling enhancing the model's adaptability to various historical distributions, thereby avoiding overfitting to a fixed sequence.

4.4 Experimental Analysis

Analysis of Time Consumption and Computational Demands.

To intuitively show the time and memory requirements of each comparison method, we detail the training epoch, memory bank size, and total time per stage alongside AVG and PRE metrics in terms of ACC. From the results in Tab. 10, it's evident that our method delivers a commendable performance with a significantly reduced memory footprint and training time compared to recent methods like FReTAL, DER++, and Co2L, all of which require 20 training epochs. Moreover, when we reduced our training epochs from 20 to 10, our proposed HDP method still managed to attain state-of-the-art performance, and in terms of time efficiency, it even surpassed SFT. These findings robustly underscore the efficiency and potency of our approach.

Regarding the computational demands, both the Reserve Mechanism and the Preserve Mechanism are employed only during training. As a result, our method does not introduce additional parameters or computational complexity during inference. The inference speed and resource consumption are solely determined by the base model and are independent

of the HDP framework. For example, if EN-B4 has 17.54M parameters and 1.54G floating-point operations, EN-B4 with HDP will exhibit identical computational performance. This advantage enhances the versatility of our method, making it applicable to various scenarios without compromising efficiency.

To quantify the computational requirements, we calculated the time needed to generate UAPs using EfficientNet-B4 (a common deepfake detection backbone) and EfficientNet-B0 (a lightweight backbone for resource-constrained devices). Generating UAPs on EN-B4 using 11,520 real images took 176 s (0.015s per image), while on EN-B0, it took 90.16s (0.007s per image), indicating low time consumption. A single UAP requires approximately 500KB \pm 100KB of storage, which is smaller than a single image and significantly less than the memory-based storage requirements of previous methods.

In summary, our method offers significant performance advantages over traditional approaches while also boasting faster convergence rates and lower resource consumption. Moreover, it does not introduce any additional inference time and requires only minimal storage space, further emphasizing its practicality and real-world applicability.

Analysis of Distribution Change. Our method intends to alleviate the catastrophic forgetting by rehearsing the simulated forgery distribution via UAP-based pseudo samples. To show the dynamic process of CFFD, we draw the feature distribution of baseline (SFT) and our method using t-SNE. Specifically, we visualize the four-step models i.e. Deepfakes, Face2Face, FaceSwap, and NeuralTextures of protocol1 with real, corresponding forgery, corresponding pseudo samples and previous forgery feature distributions, respectively. The results are shown in Fig. 4. We can observe that 1) for the SFT model, the overlap between previous forgery features and real features becomes more and more obvious with the increase of the attack, which demonstrates the phenomenon of catastrophic forgetting. 2) For our method, there exists an obvious decision boundary between the previous forgery feature and the real feature in every stage, which demonstrates the effectiveness of the anti-forgetting of our method. 3) The distribution of pseudo-forged features and actual forgery features is resembling and overlapping with each other. This phenomenon can support the motivation that uses these pseudo samples to simulate the forgery distribution.

5 Practicals of CFFD

As AIGC advances, the variety of manipulated faces increases, making it impossible for the detection model to acquire all types of training data at once. This results in poor performance of the model when confronted with unfamiliar domain

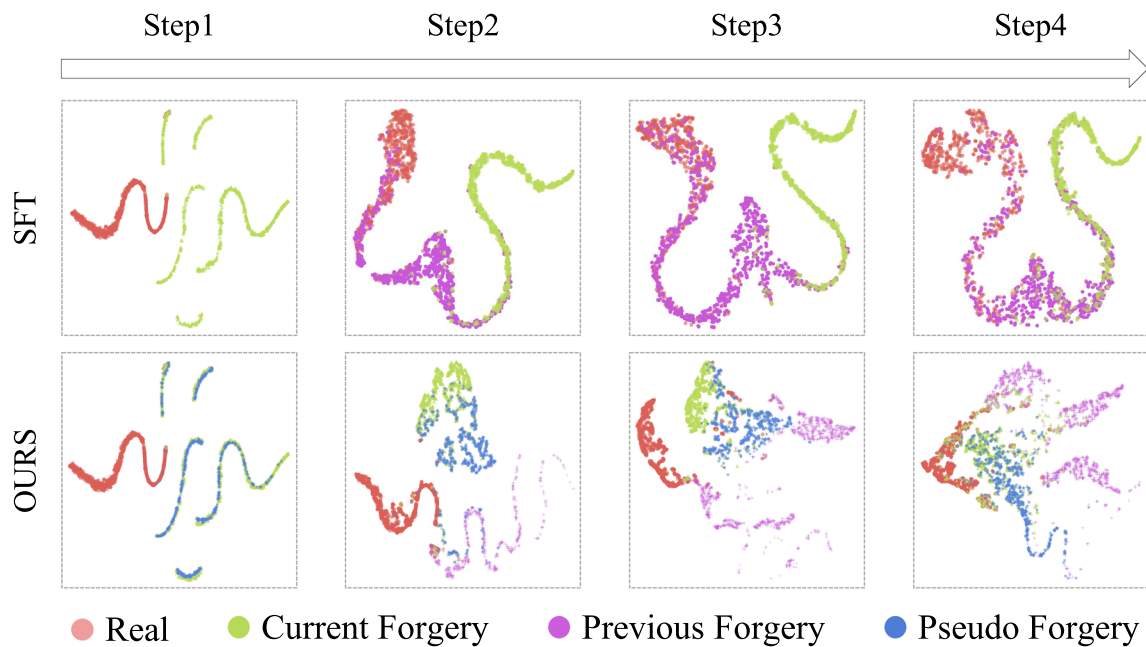


Fig. 4 Visualization of feature distribution on protocol1 via t-SNE. Step1 to step4 represents the Deepfakes, Face2Face, FaceSwap and Neural-Textures, respectively

data, which severely constrains the deployment of the detection model. Although some methods have been proposed to enhance the generalization of the model, even the SOTA generalization models are still far from practical application. For instance, on the Celeb-DF dataset, the best generalization method achieves an AUC of 84% under the cross-domain setting, while the intra-domain performance is 99% using the same backbone. Therefore, in real-world applications, the detection model should be updated along with the emergence of new forgery methods. A naive way to update the model is to combine the newly collected data and historical data for training, but this has two drawbacks: first, it will consume a lot of computational and storage resources; second, it will cause security issues such as data leakage. Therefore, in this paper, we focus on the problem of continual face forgery detection (CFFD), which aims to enable the model to learn new data and avoid forgetting the previous knowledge simultaneously. We believe that the setting of CFFD will contribute to the deployment and promotion of face forgery detection models.

6 Conclusion

In this work, we focus on a challenging and practical setting, named Continual Face Forgery face detection (CFFD), which enables efficient learning to deal with continually new attacks while mitigating catastrophic forgetting. Specifically, we propose a novel historical distribution preserving

framework, which preserves the previous forgery distribution based on Universal Adversarial Perturbation (UAP), and the knowledge distillation is further introduced to maintain distribution variation of real faces across models over different periods. Correspondingly, we also build a new benchmark for CFFD with three different evaluation protocols. Extensive experiments on the novel benchmarks demonstrate that our method outperforms other SOTA competitors at a lower storage cost. The results affirm the potential of our approach as a robust and efficient solution in the ongoing battle against face forgery and underline the significance of continual learning in cybersecurity.

Appendix A: Analysis of UAP Pseudo Forgery Samples

Additional Visualization of UAP We also provide more visualization results of UAP for the different stages of protocol1 and protocol2. Furthermore, we have added the visualization of UAPs corresponding to AIGC-related methods in the third row. We can observe from Fig. 5 that different stages have their own discriminative UAP patterns.

Analysis the Distribution of Pseudo Forgery Samples

Apparently, our strategy relies on a hypothesis that the UAP based pseudo forgery samples should have the same distribution as the actual forgery samples. To verify this conjecture, we illustrate the feature distributions of four well-trained face

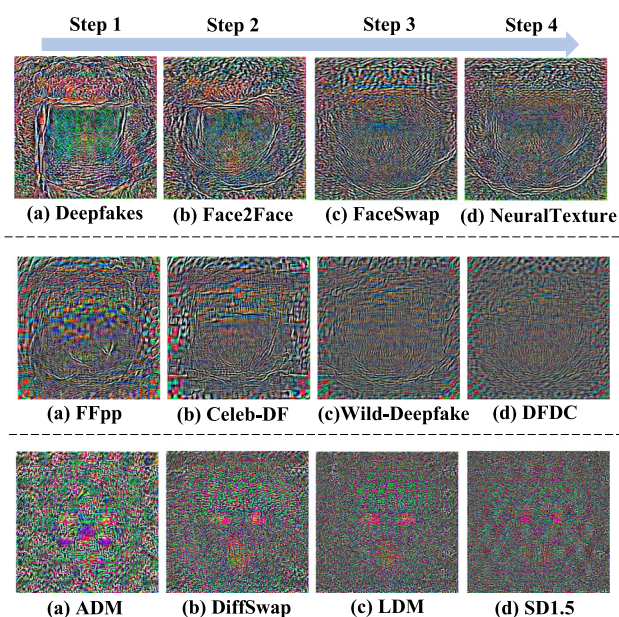


Fig. 5 UAPs visualization for different attack methods. The first two rows represent traditional deepfake methods, while the last row corresponds to UAPs generated by AIGC methods

forgery detection models. Specifically, we first train each model on subsets of FFpp (Rossler et al., 2019) i.e. Deepfakes, Face2Face, FaceSwap and NeuralTextures using Eq. 1. Then we generate respective UAP under the guidance of reserve mechanism details in Sect. 3.4. Finally, we present the feature distribution of real, forgery and pseudo forgery by t-SNE, as shown in Fig. 6. We can observe that the distribution of pseudo-forged samples almost overlaps with the forgery features across all four datasets, which demonstrates the high similarity between the two distributions. To explain, as stated in previous works (Zhang et al., 2020; Jetley et al., 2018), UAP are **features** that contains fixed directional perturbations across class boundaries irrespective of the diversity in their individual appearance. Jetley et al. (2018) also demonstrate the close relationship between certain directions and class identity, i.e. the network identifies the certain directions as features associated with class identities. Therefore, we argue that the pseudo-forged samples of UAP can maintain the distribution of the forgery class and it is intuitive to use it as the surrogate of actual forgery feature distribution.

Appendix B: Additional Experimental Results

Result on Protocol 1 with Order2 Since the stage order is agnostic for lifelong learning, we report the details of protocol 1 with *order2*, that is **NT** → **DF** → **F2F** → **FS**. The results are shown in Tab. 11. Our proposed HDP significantly outperforms all comparing methods in terms of both AVG and PRE. Compared with baseline SFT, our method

achieve 25% performance gain in PRE due to the historical distribution preservation. Furthermore, compared with the latest replayed methods such as Co2L and FReTAL required additional buffer size, our method alleviates the catastrophic forgetting by reserving only one UAP per stage. The SOTA results on both *order1* and *order2* demonstrate our method is insensitive to the stage order.

Result on Protocol 1 with AIGC To explore the generalizability of our method against the latest manipulation methods, we created a continual benchmark related to face AIGC using the latest Diffusion (Ho et al., 2020) technology. Specifically, we use DiffusionFace (Chen et al., 2024) dataset which employed the Multi-Modal-CelebA-HQ (Xia et al., 2021) dataset as the real face data and generated corresponding fake faces using four advanced Diffusion-based generation methods: ADM (Dhariwal & Nichol, 2021), DiffSwap (Zhao et al., 2023), LDM (Rombach et al., 2022), and StableDiffusion 1.5. We then evaluated all the compared methods following Protocol 1. The quantitative results in Tab. 12 show that although the in-domain performance is high, the SFT suffers from severe catastrophic forgetting, with a higher forgetting rate than traditional Deepfake methods, achieving a PRE of 39.17% in terms of ACC. In contrast, our HDP method can improve the PRE by 22% in terms of ACC, demonstrating the scalability and robustness of our approach when faced with such new attacks. Furthermore, compared to other continual learning methods, our method exhibits significant advantages. Specifically, our method achieves a 4% improvement in AVG compared to the current state-of-the-art PEC method. Moreover, our method attains SOTA performance without using a memory bank, unlike other methods that require one, proving the effectiveness of UAP as a feature. Figure 5 illustrates the visualization of the UAP generated from the AIGC dataset. This experiment demonstrates the generalizability of our method when confronted with the most advanced forgery techniques available today.

Result on Protocol 1 with Unseen Metric In addition to the CFFD setting, which can alleviate the poor generalization issue of current methods to a certain extent, our HDP framework also provides performance gains for unseen attacks. To quantify this, we introduce the Unseen metric, which represents the average performance of the model on attack methods that have not been encountered during the training of each stage in Protocol 1. Specifically, for each stage, we calculate the model's performance on the attack methods that will be introduced in future stages and then average these values across all stages. This metric effectively captures the model's ability to generalize to unseen attack methods. Tab. 13 presents the Unseen metric for our method and other comparative continual learning methods. Compared to the baseline SFT method, our HDP achieves a 4.49% improve-

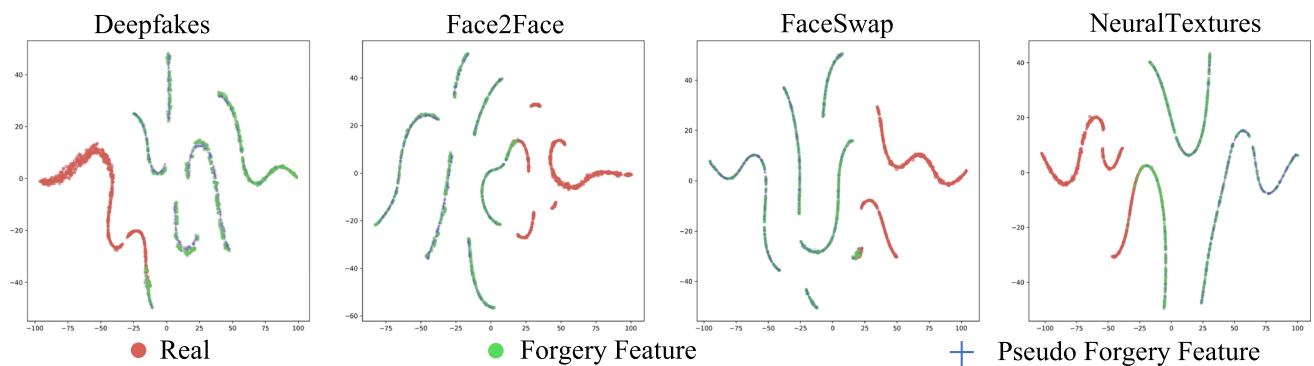


Fig. 6 Visualization of the feature distribution of four well-trained models via t-SNE. The **red** dot represents the real features. The **green** triangle is the corresponding actual forgery features. The **blue** cross denotes the pseudo forgery features based UAP (Color figure online)

Table 11 Quantitative results on protocol1 with order2 in terms of ACC and AUC

Method	Buffer/step	NeuralTextures		Deepfakes		Face2Face		FaceSwap		AVG		PRE	
		ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
SFT	0	58.01	67.31	64.09	88.02	64.78	94.55	98.00	99.64	71.22	87.38	64.19	85.51
LwF	0	58.24	86.22	70.12	95.45	71.17	96.50	98.76	99.69	74.57	94.46	70.18	93.33
MAS	0	76.58	89.86	74.00	88.48	74.74	89.17	95.62	98.82	80.23	91.58	80.99	92.30
NCM	0	58.02	57.13	66.71	73.00	62.95	69.97	99.06	99.19	71.68	74.82	63.98	68.73
iCaRL	500	<u>77.32</u>	89.89	83.08	93.37	82.20	96.18	98.86	99.71	85.36	96.53	80.73	94.78
SCR	500	75.88	<u>93.43</u>	<u>84.49</u>	<u>96.37</u>	<u>88.72</u>	<u>97.91</u>	98.74	99.56	<u>86.95</u>	<u>96.76</u>	<u>84.12</u>	<u>95.22</u>
FReTAL	500	71.99	86.40	81.94	93.85	87.74	97.35	<u>99.04</u>	99.62	83.85	94.78	82.20	93.71
DER	500	58.00	66.55	81.67	94.72	68.72	91.13	98.74	<u>99.72</u>	76.78	88.03	69.95	82.23
DER++	500	60.03	47.78	71.26	79.21	62.27	70.39	98.80	99.37	73.09	74.18	66.20	65.85
Co2L	500	58.21	65.30	73.48	90.29	69.76	90.89	99.08	99.73	75.13	86.55	67.50	83.62
HDP	0	80.36	94.27	91.56	98.06	94.60	98.50	98.52	<u>99.06</u>	91.29	97.47	89.80	97.35
Joint	—	94.04	97.84	96.99	99.75	96.35	99.12	96.63	99.30	96.00	99.00	—	—

We evaluate the last model on all the previous test sets. The buffer represents the memory bank budget for each stage. The underlined values represent the second best result. The values in bold represent the best results

ment in the Unseen AUC metric. Furthermore, our method obtains state-of-the-art results compared to other continual learning methods. This can be attributed to our approach of using UAP for data augmentation when training on current attack data, which enhances the network's ability to learn a broader forgery distribution and capture more essential forgery features. This, in turn, promotes the model's generalization to unseen attacks. These results further demonstrate the effectiveness of our HDP method against adaptive adversaries and in scenarios with undefined attack vectors.

Appendix C: Protocols Details

In this section, we elucidate the composition of sub-datasets for the three proposed protocols within the Continual Face Forgery Detection (CFFD). Protocol 1 comprises four subsets from FFpp, which are specifically curated to simulate

the ever-evolving new attack methods. In contrast, Protocol 2 encompasses four separate face forgery datasets. These datasets introduce substantial domain variations for both authentic and counterfeit images across each phase. This structure results in a richer assortment of attack types at every stage compared to Protocol 1, which focuses on a singular attack type in each phase. Meanwhile, Protocol 3 is formulated to gauge the performance of models when confronted with prolonged sequences of emerging genuine and doctored images. This protocol is segmented into 10 stages, integrating four subsets from FFpp along with evenly segmented versions of the Celeb-DF, Wild-Deepfake, and DFDC datasets. The intricate design of Protocol 3 accentuates the challenge posed by the catastrophic forgetting phenomenon. Through a deeper comprehension of these protocols' distinct compositions and objectives, researchers are better equipped to devise and assess face forgery detection mechanisms.

Table 12 Quantitative results on AIGC dataset in terms of ACC and AUC

Method	Buffer/step	ADM		DiffSwap		LDM		StableDiffusion		AVG		PRE	
		ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
SFT	0	33.36	53.58	33.36	57.29	33.49	89.33	99.97	99.99	50.04	75.05	39.17	79.06
LwF	0	41.27	57.91	40.31	61.25	39.29	90.12	99.92	99.98	55.19	77.31	45.28	83.50
MAS	0	49.96	70.99	48.30	71.59	45.22	90.99	96.37	98.18	59.96	82.93	53.25	86.67
NCM	0	42.79	60.11	39.23	59.91	38.57	89.95	98.33	98.89	54.73	77.21	49.17	82.20
PRKA	0	48.29	75.51	50.12	70.25	52.38	91.13	99.92	99.79	62.67	84.17	56.60	86.14
PEC	0	46.71	72.53	51.17	71.25	55.39	92.73	99.96	99.92	63.30	84.10	57.02	84.21
iCaRL	500	49.29	77.25	53.17	72.70	60.55	93.31	99.96	99.97	65.74	85.80	54.64	88.34
SCR	500	50.33	78.50	52.14	71.11	62.23	94.31	99.93	99.91	66.15	85.95	56.73	89.15
FReTAL	500	51.05	81.25	54.47	73.15	63.55	94.13	99.95	99.97	67.25	87.12	58.65	90.23
DER++	500	51.27	84.57	55.28	74.58	65.29	94.19	99.96	99.98	67.95	88.33	58.77	89.93
Foster	500	51.95	85.03	54.38	73.25	66.14	94.57	99.95	99.93	68.10	88.19	59.09	90.99
HDP	0	53.39	87.48	57.96	77.51	67.28	95.95	99.96	99.98	69.64	90.23	61.24	93.15
Joint	–	99.80	99.99	99.76	99.99	99.81	99.99	99.80	99.99	99.79	99.99	–	–

We evaluate the last model on all the previous test sets. The buffer represents the memory bank budget save the previous images per stage. The values in bold represent the best results

Table 13 Quantitative results on protocol1 in terms of ACC and AUC of Unseen metric

Method	Buffer/step	Unseen	
		ACC	AUC
SFT	0	58.68	68.60
LwF	0	58.71	71.93
MAS	0	59.46	70.31
NCM	0	62.84	70.97
PRKA	0	61.74	69.95
PEC	0	61.98	70.04
iCaRL	500	59.33	70.92
SCR	500	58.68	68.36
FReTAL	500	60.89	71.56
DER++	500	58.95	67.81
Co2L	500	59.20	66.55
Foster	500	61.22	70.59
HDP	0	63.10	73.09

The Unseen metric represents the average performance of each stage. The buffer represents the memory bank budget save the previous images per stage. The values in bold represent the best results

Appendix D: Practicals of CFFD

Difference Among Continual Learning, Domain Generalization and Few-Shot Learning In the realm of face forgery detection, several research directions have emerged that bear similarities to continual learning, notably domain generalization and few-shot learning. This section delineates the distinctions between Continual Face Forgery Detection

(CFFD) and these methodologies. Few-shot learning aims to optimize a model's performance on a test set using a minimal amount of training data. The primary focus here is to leverage this scant data to its fullest potential. Domain generalization, on the other hand, trains models using a known source domain and then evaluates their performance on an entirely unseen target domain. The objective is to enhance the model's capacity to distinguish data it has never encountered before. Continual learning has a distinct goal: to incorporate and adapt to new data streams without erasing or overshadowing previously acquired knowledge. This approach is particularly concerned with avoiding “catastrophic forgetting” when updating with ample new training data.

Exploring the synergies and integrations between CFFD, domain generalization, and few-shot learning promises to be a captivating avenue for future investigations. Domain generalization focuses on learning representations that are robust and transferable across different domains, while few-shot learning aims to quickly adapt to new tasks with limited training examples. Combining these approaches with CFFD could lead to the development of more versatile and adaptable face forgery detection systems. For instance, by leveraging domain generalization techniques, CFFD models could be trained to capture more domain-invariant features, enabling them to generalize better to unseen forgery techniques. Similarly, incorporating few-shot learning strategies could allow CFFD models to rapidly adapt to new forgery methods with only a few examples, reducing the need for extensive retraining.

Acknowledgements This work was partially supported by National Science and Technology Major Project (No. 2022ZD0118202), the

National Science Fund for Distinguished Young Scholars (No. 62025603), the National Natural Science Foundation of China (Nos. U21B2037, U22B2051, 62176222, 62176223, 62176226, 62072386, 62072387, 62072389, 62002305 and 62272401), and the Natural Science Foundation of Fujian Province of China (Nos. 2021J01002 and 2022J06001).

Data Availability Statement The data from four public face forgery detection datasets (i.e., FaceForensics++ (Rossler et al., 2019), Celeb-DF (Li et al., 2020), DFDC (Dolhansky et al., 2020), Wild-Deepfake (Zi et al., 2020), Multi-Modal-CelebA-HQ (Xia et al., 2021)) These datasets have been instrumental in advancing research in the field of face forgery detection and have been widely used by the research community. Such datasets support the findings of this study and are available from third party institutions but restrictions apply to the availability of these data, which were used under license for the current study, and so are not publicly available. Data are however available from the authors upon reasonable request and with permission of the above-mentioned third-party institutions. This allows the datasets to be used responsibly and ethically while still allowing the scientific community to build upon the findings of this study and advance the state-of-the-art in face forgery detection.

References

- Afchar, D., Nozick, V., Yamagishi, J., & Echizen, I. (2018). Mesonet: A compact facial video forgery detection network. In *Wifs, IEEE* (pp. 1–7).
- Aljundi, R., Babiloni, F., Elhoseiny, M., Rohrbach, M., & Tuytelaars, T. (2018). Memory aware synapses: Learning what (not) to forget. In *ECCV* (pp. 139–154).
- Aljundi, R., Lin, M., Goujaud, B., & Bengio, Y. (2019). Online continual learning with no task boundaries. *arXiv preprint arXiv:1903.08671*.
- Buzzega, P., Boschini, M., Porrello, A., Abati, D., & Calderara, S. (2020). Dark experience for general continual learning: A strong, simple baseline. *NeurIPS*, 33, 15920–15930.
- Cha, H., Lee, J., & Shin, J. (2021). Co2l: Contrastive continual learning. In *ICCV* (pp. 9516–9525).
- Chaubey, A., Agrawal, N., Barnwal, K., Guliani, K. K., & Mehta, P. (2020). Universal adversarial perturbations: A survey. *arXiv preprint arXiv:2005.08087*.
- Chaudhry, A., Gordo, A., Dokania, P. K., Torr, P., & Lopez-Paz, D. (2020). Using hindsight to anchor past knowledge in continual learning. *arXiv preprint arXiv:2002.08165*.
- Chen, Z., Sun, K., Zhou, Z., Lin, X., Sun, X., Cao, L., & Ji, R. (2024). Diffusionface: Towards a comprehensive dataset for diffusion-based face forgery analysis. *arXiv preprint arXiv:2403.18471*.
- Chen, S., Yao, T., Chen, Y., Ding, S., Li, J., & Ji, R. (2021). Local relation learning for face forgery detection. In *AAAI*.
- Dang, H., Liu, F., Stehouwer, J., Liu, X., & Jain, A. K. (2020). On the detection of digital face manipulation. In *CVPR* (pp. 5781–5790).
- Delange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., & Tuytelaars, T. (2021). A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *CVPR, IEEE* (pp. 248–255).
- Dhariwal, P., & Nichol, A. (2021). Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems*, 34, 8780–8794.
- Dolhansky, B., Bitton, J., Pflaum, B., Lu, J., Howes, R., Wang, M., & Ferrer, C. C. (2020). The deepfake detection challenge dataset. *arXiv preprint arXiv:2006.07397*.
- Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33, 6840–6851.
- Hou, S., Pan, X., Loy, C. C., Wang, Z., & Lin, D. (2019). Learning a unified classifier incrementally via rebalancing. In *CVPR* (pp. 831–839).
- Jetley, S., Lord, N., & Torr, P. (2018). With friends like these, who needs adversaries? *NeurIPS* 31.
- Kim, M., Tariq, S., & Woo, S. S. (2021). Fretal: Generalizing deepfake detection using knowledge distillation and representation learning. In *CVPR* (pp. 1001–1012).
- Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., & Grabska-Barwinska, A. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13), 3521–3526.
- Li, L., Bao, J., Zhang, T., Yang, H., Chen, D., Wen, F., & Guo, B. (2020). Face x-ray for more general face forgery detection. In *CVPR*, pp. 5001–5010.
- Li, C., Huang, Z., Paudel, D. P., Wang, Y., Shahbazi, M., Hong, X., & Van Gool, L. (2023). A continual deepfake detection benchmark: Dataset, methods, and essentials. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 1339–1349).
- Li, J., Wang, Y., Wang, C., Tai, Y., Qian, J., Yang, J., Wang, C., Li, J., & Huang, F. (2019). DSFD: Dual shot face detector. In *CVPR*, pp. 5060–5069.
- Li, Y., Yang, X., Sun, P., Qi, H., & Lyu, S. (2020). Celeb-DF: A large-scale challenging dataset for deepfake forensics. In *CVPR* pp. 3207–3216.
- Li, Z., & Hoiem, D. (2017). Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12), 2935–2947.
- Liu, H., Ji, R., Li, J., Zhang, B., Gao, Y., Wu, Y., & Huang, F. (2019). Universal adversarial perturbation via prior driven uncertainty approximation. In *ICCV* (pp. 2941–2949).
- Luo, A., Li, E., Liu, Y., Kang, X., & Wang, Z. J. (2021). A capsule network based approach for detection of audio spoofing attacks. In *ICASSP 2021-2021 IEEE international conference on acoustics, speech and signal processing (ICASSP) IEEE* (pp. 6359–6363).
- Luo, Y., Zhang, Y., Yan, J., & Liu, W. (2021). Generalizing face forgery detection with high-frequency features. In *CVPR*, (pp. 16317–16326).
- Luo, A., Kong, C., Huang, J., Hu, Y., Kang, X., & Kot, A. C. (2023). Beyond the prior forgery knowledge: Mining critical clues for general face forgery detection. *IEEE Transactions on Information Forensics and Security*, 19, 1168–1182.
- Mai, Z., Li, R., Kim, H., & Sanner, S. (2021). Supervised contrastive replay: Revisiting the nearest class mean classifier in online class-incremental continual learning. In *CVPR* (pp. 3589–3599).
- Matern, F., Riess, C., & Stamminger, M. (2019). Exploiting visual artifacts to expose deepfakes and face manipulations. In *WACVWIEEE*, (pp. 83–92).
- Moosavi-Dezfooli, S.-M., Fawzi, A., & Frossard, P. (2016). Deepfool: A simple and accurate method to fool deep neural networks. In *CVPR* (pp. 2574–2582).
- Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., & Frossard, P. (2017). Universal adversarial perturbations. In *CVPR* pp. 1765–1773.
- Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., Frossard, P., & Soatto, S. (2018). Robustness of classifiers to universal perturbations: A geometric perspective. In *ICLR*.
- Qian, Y., Yin, G., Sheng, L., Chen, Z., & Shao, J. (2020). Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *ECCV*, (pp. 86–103). Springer: Berlin

- Rebuffi, S.-A., Kolesnikov, A., Sperl, G., & Lampert, C. H. (2017). ICARL: Incremental classifier and representation learning. In *CVPR* (pp. 2001–2010).
- Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 10684–10695).
- Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). Faceforensics++: Learning to detect manipulated facial images. In *ICCV* (pp. 1–11).
- Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., & Hadsell, R. (2016). Progressive neural networks. arXiv preprint [arXiv:1606.04671](https://arxiv.org/abs/1606.04671)
- Shi, W., & Ye, M. (2023). Prototype reminiscence and augmented asymmetric knowledge aggregation for non-exemplar class-incremental learning. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 1772–1781).
- Stehouwer, J., Dang, H., Liu, F., Liu, X., & Jain, A. (2019). On the detection of digital face manipulation. arXiv preprint [arXiv:1910.01717](https://arxiv.org/abs/1910.01717)
- Sun, K., Chen, S., Yao, T., Sun, X., Ding, S., & Ji, R. (2023). Towards general visual-linguistic face forgery detection. arXiv preprint [arXiv:2307.16545](https://arxiv.org/abs/2307.16545)
- Sun, K., Liu, H., Yao, T., Sun, X., Chen, S., Ding, S., & Ji, R. (2022). An information theoretic approach for attention-driven face forgery detection. In: *ECCV* (pp. 111–127). Springer: Berlin.
- Sun, K., Liu, H., Ye, Q., Liu, J., Gao, Y., Shao, L., & Ji, R. (2021). Domain general face forgery detection by learning to weight. In *AAAI* (Vol. 35, pp. 2638–2646).
- Sun, K., Yao, T., Chen, S., Ding, S., & Ji, R. (2022). Dual contrastive learning for general face forgery detection. In: *AAAI*
- Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing properties of neural networks. arXiv preprint [arXiv:1312.6199](https://arxiv.org/abs/1312.6199)
- Tan, M., & Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*.
- Thies, J., Zollhöfer, M., Nießner, M., Valgaerts, L., Stamminger, M., & Theobalt, C. (2015). Real-time expression transfer for facial reenactment. *ACM Trans. Graph.*, 34(6), 183–1.
- Tung, F., & Mori, G. (2019). Similarity-preserving knowledge distillation. In *ICCV* (pp. 1365–1374).
- Verma, V. K., Liang, K. J., Mehta, N., Rai, P., & Carin, L. (2021). Efficient feature transformations for discriminative and generative continual learning. In *CVPR*, pp. 13865–13875
- Wang, Z., Zhang, Z., Lee, C.-Y., Zhang, H., Sun, R., Ren, X., Su, G., Perot, V., Dy, J., & Pfister, T. (2021). Learning to prompt for continual learning. arXiv preprint [arXiv:2112.08654](https://arxiv.org/abs/2112.08654)
- Wang, F.-Y., Zhou, D.-W., Ye, H.-J., Zhan, D.-C. (2022). Foster: Feature boosting and compression for class-incremental learning. In: *European Conference on Computer Vision* (pp. 398–414). Springer: Berlin
- Wu, Y., Chen, Y., Wang, L., Ye, Y., Liu, Z., Guo, Y., & Fu, Y. (2019). Large scale incremental learning. In *CVPR* (pp. 374–382).
- Xia, W., Yang, Y., Xue, J.-H., & Wu, B. (2021). Tedigan: Text-guided diverse face image generation and manipulation. In *IEEE conference on computer vision and pattern recognition (CVPR)*.
- Xu, J., & Zhu, Z. (2018). Reinforced continual learning. *NeurIPS* 31.
- Yan, S., Xie, J., & He, X. (2021). Der: Dynamically expandable representation for class incremental learning. In *CVPR* (pp. 3014–3023).
- Yan, Z., Zhang, Y., Fan, Y., & Wu, B. (2023). UCF: Uncovering common features for generalizable deepfake detection. In *ICCV*.
- Yang, X., Li, Y., & Lyu, S. (2019). Exposing deep fakes using inconsistent head poses. In *ICASSP, IEEE* (pp. 8261–8265).
- Zajac, M., Tuytelaars, T., & Ven, G. M. (2024). Prediction error-based classification for class-incremental learning. In *ICLR*.
- Zhang, C., Benz, P., Imtiaz, T., & Kweon, I. S. (2020). Understanding adversarial examples from the mutual influence of images and perturbations. In *CVPR*, (pp. 14521–14530).
- Zhao, W., Rao, Y., Shi, W., Liu, Z., Zhou, J., & Lu, J. (2023). Diffswap: High-fidelity and controllable face swapping via 3d-aware masked diffusion. In *CVPR*.
- Zhao, T., Xu, X., Xu, M., Ding, H., Xiong, Y., & Xia, W. (2021). Learning self-consistency for deepfake detection. In *CVPR*, pp. 15023–15033.
- Zhao, H., Zhou, W., Chen, D., Wei, T., Zhang, W., & Yu, N. (2021). Multi-attentional deepfake detection. In *CVPR*.
- Zhuang, W., Chu, Q., Tan, Z., Liu, Q., Yuan, H., Miao, C., Luo, Z., & Yu, N. (2022). UIA-VIT: Unsupervised inconsistency-aware method based on vision transformer for face forgery detection. In: *European Conference on Computer Vision*, pp. 391–407. Springer: Berlin
- Zi, B., Chang, M., Chen, J., Ma, X., & Jiang, Y.-G. (2020). Wilddeepfake: A challenging real-world dataset for deepfake detection. In *ACM MM* (pp. 2382–2390).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.