Unifying Locality of KANs and Feature Drift Compensation Projection for Data-Free Replay-Based Continual Face Forgery Detection

Tianshuo Zhang[1,2], **Siran Peng**[1,2], **Li Gao**[3], **Haoyuan Zhang**[1,2], **Xiangyu Zhu**[1,2], **Zhen Lei**[1,2,4,5*]

[1]School of Artificial Intelligence, University of Chinese Academy of Sciences
[2]MAIS, Institute of Automation, Chinese Academy of Sciences
[3]China Mobile Financial Technology Co., Ltd.
[4]CAIR, HKISI, Chinese Academy of Sciences
[5]SCSE, the Faculty of Innovation Engineering, M.U.S.T
tianshuo.zhang@nlpr.ia.ac.cn, {pengsiran2023, zhanghaoyuan2023, xiangyu.zhu, zhen.lei}@ia.ac.cn,
gaolids@chinamobile.com

Abstract

The rapid advancements in face forgery techniques necessitate that detectors continuously adapt to new forgery methods, thus situating face forgery detection within a continual learning paradigm. However, when detectors learn new forgery types, their performance on previous types often degrades rapidly, a phenomenon known as catastrophic forgetting. Kolmogorov-Arnold Networks (KANs) utilize locally plastic splines as their activation functions, enabling them to learn new tasks by modifying only local regions of the functions while leaving other areas unaffected. Therefore, they are naturally suitable for addressing catastrophic forgetting. However, KANs have two significant limitations: 1) the splines are ineffective for modeling high-dimensional images, while alternative activation functions that are suitable for images lack the essential property of locality; 2) in continual learning, when features from different domains overlap, the mapping of different domains to distinct curve regions always collapses due to repeated modifications of the same regions. In this paper, we propose a **KAN**-based **C**ontinual Face **F**orgery **D**etection (KAN-CFD) framework, which includes a Domain-Group KAN Detector (DG-KD) and a data-free replay Feature Separation strategy via KAN Drift Compensation Projection (FS-KDCP). DG-KD enables KANs to fit high-dimensional image inputs while preserving locality and local plasticity. FS-KDCP avoids the overlap of the KAN input spaces without using data from prior tasks. Experimental results demonstrate that the proposed method achieves superior performance while notably reducing forgetting.

Introduction

The rapid development of AI-generated content (AIGC) accelerates the evolution of face manipulation methods, making it challenging for the public to discern forged media and raising significant societal concerns. Static models (Nguyen et al. 2024; Peng et al. 2025; Yan et al. 2025) trained on fixed datasets exhibit limited generalization capabilities, rendering them ineffective against these rapidly evolving forgery techniques. Continual learning methods (Yan, Xie, and He 2021; Shin et al. 2017; Kirkpatrick et al. 2017) enable a model to be continuously updated over a sequence of tasks, representing a key strategy for detection models to keep pace with the evolution of forgery methods. However, when learning new tasks, these models often suffer a severe performance drop on previous tasks, a phenomenon known as catastrophic forgetting. Most continual face forgery detection models (Pan et al. 2023; Cheng et al. 2025) mitigate this issue by retaining and replaying a set of data from prior tasks, but this approach introduces additional privacy risks.

Kolmogorov-Arnold Networks (KANs) (Liu et al. 2024) replace the linear weights and global activation functions of MLPs (Hornik, Stinchcombe, and White 1989) with learnable activation functions located on the edges of the

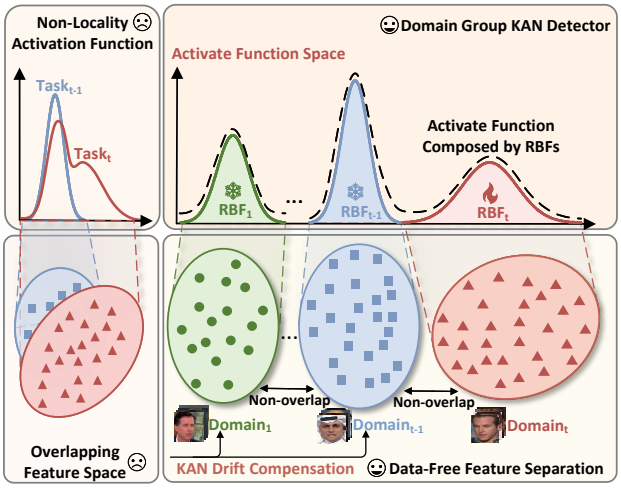


Figure 1: Feature space overlap and non-local activations in image-KANs hinder continual learning. We propose DG-KD, which constructs KAN learnable activation functions using combined RBFs to preserve both locality and image fitting capabilities. Meanwhile, FS-KDCP separates feature distributions without data replay. Their synergy ensures learning of different tasks without mutual interference.

*Corresponding author.

network, demonstrating superior performance compared to MLPs across a wide range of applications (Park, Kim, and Shin 2024; Bodner et al. 2024). Specifically, the learnable activation functions in KANs exhibit locality and local plasticity. In sequential tasks, this property enables the network to adjust local curves to acquire knowledge of new tasks while preserving stability in other regions, making KANs a naturally suitable architecture for continual learning.

However, two primary limitations impede the application of KANs to continual learning scenarios. First, the original B-spline activation functions are computationally expensive, and their basis functions are ill-suited for modeling the high-dimensional data distributions characteristic of vision tasks. Several existing methods offer efficient alternatives for this activation function (Buhmann 2000; Aghaei 2024). However, they all lack the property of locality that is essential for continual learning (Hu et al. 2025). Second, (Lee et al. 2025) demonstrates that in sequential tasks, overlapping feature distributions from different tasks result in repeated modifications of the same spline regions, which lead to the erasure of knowledge acquired from previous tasks. This issue becomes particularly severe in continual face forgery detection, where the classification objective causes the feature distributions of different domains to significantly overlap.

In this paper, we introduce a KAN-based framework for Continual Face Forgery Detection (KAN-CFD) to address the aforementioned limitations. As illustrated in Fig. 1, we first propose a novel Domain-Group KAN Detector (DG-KD) that utilizes a combination of domain-specific RBFs to form DG-Layers, leveraging RBFs to enable the fitting of high-dimensional image inputs. By combining locally non-zero RBFs from different regions, it provides the essential properties of locality and plasticity required for continual learning. Second, we introduce a data-free replay-based Feature Separation strategy via KAN Drift Compensation Projection (FS-KDCP) to ensure a non-overlapping input space for DG-KD. FS-KDCP employs a KAN projection (KDCP) to model the feature drift of the backbone and transform stored features into the current feature space to compensate for this drift. These transformed features are then used to estimate the overall distributions of each previous domain. Finally, the strategy separates the feature distributions of the new and old tasks to yield a non-overlapping feature space. This approach allows different domains to be modeled in distinct regions of DG-KD without requiring data from prior tasks. In summary, our main contributions are:

- We propose the Domain-Group KAN Detector (DG-KD), which leverages a combination of domain-specific RBFs to model high-dimensional image inputs while providing the essential locality and local plasticity.
- We introduce a data-free replay-based Feature Separation strategy via KAN Drift Compensation Projection (FS-KDCP), creating a non-overlapping input space for the DG-KD without requiring data from prior tasks.
- Extensive experiments demonstrate that our proposed framework establishes a new state-of-the-art (SOTA), achieving superior detection accuracy and the lowest forgetting rate compared to all previous methods.

Related Works

Kolmogorov-Arnold Networks

Kolmogorov-Arnold Networks (KANs) (Liu et al. 2024) demonstrate superior performance compared to traditional MLPs across a wide range of tasks. However, the original KAN architecture is not well suited for vision applications due to challenges in modeling high-dimensional data. To enhance efficiency, subsequent research, including rKAN (Aghaei 2024) and FastKAN (Li 2024), investigates the substitution of B-splines with rational basis functions, while KAT (Xingyi Yang 2025) introduces the Group-KAN architecture, which utilizes shared activation functions across different dimensions. In the field of continual learning, WiseKAN (Lee et al. 2025) reveals that feature overlap in the KAN input space leads to catastrophic forgetting. KAC (Hu et al. 2025) presents a KAN classifier that achieves high performance in Class-Incremental Learning (CIL).

Continual Face Forgery Detection

Continual face forgery detection is typically formulated as a domain-incremental learning problem. Prevailing approaches primarily rely on knowledge distillation and data replay. For instance, CoReD (Kim, Tariq, and Woo 2021) introduces a distillation loss to preserve task-specific knowledge. DFIL (Pan et al. 2023) proposes a novel data selection strategy for replay and achieves competitive performance by employing distillation on both feature and label representations. DMP (Tian et al. 2024) introduces prototype learning and a prototype-guided replay strategy to preserve knowledge of past tasks. HDP (Sun et al. 2025) utilizes Universal Adversarial Perturbations to replay fake samples, while SUR-LID (Cheng et al. 2025) achieves high performance by employing a separation and alignment strategy.

Methodology

Preliminaries

Continual Face Forgery Detection. We present forgery detection as a problem of domain incremental learning, denoted as $\mathcal{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_T\}$, with T representing the number of tasks. For each task $\mathcal{D}_t = \{(\boldsymbol{x}_{i,t}, y_{i,t})\}_{i=1}^{n_t} = \{\mathcal{X}_t, \mathcal{Y}_t\}$, $\boldsymbol{x}_{i,t} \in \mathcal{X}_t$ represents the samples, and $y_{i,t} \in \mathcal{Y}_t$ denotes their labels, with n_t being the number of examples in the task. The model processes each task $\mathcal{D}_t$ to learn how to handle novel forms of forgery. In the context of continual face forgery detection, tasks are characterized by varying data distributions but consistently utilize two labels:

$$p\left(\mathcal{X}_i\right) \neq p\left(\mathcal{X}_j\right) \text{ for } \mathcal{Y}_i = \mathcal{Y}_j = \text{Real or Fake and } i \neq j, \tag{1}$$

where $p\left(\mathcal{X}_i\right)$ denotes the sample distribution of task $\mathcal{D}_i$.

KAN Preliminaries. The Kolmogorov-Arnold Representation Theorem (Kolmogorov 1961) posits that any multivariate continuous function $f(\boldsymbol{x})$ defined on a bounded domain can be represented as a finite composition of univariate continuous functions combined through addition:

$$f(\boldsymbol{x}) = f(x_1, x_2, ..., x_n) = \sum_{q=1}^{2n+1} \Phi_q \Big(\sum_{p=1}^{n} \phi_{q,p}(x_p) \Big), \tag{2}$$

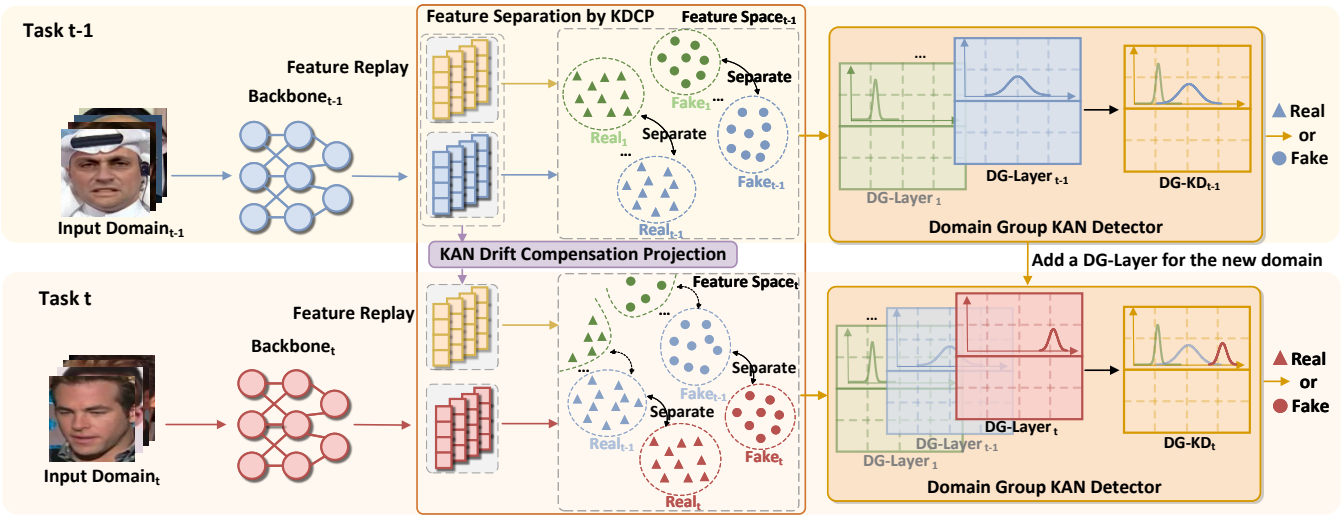


Figure 2: Our KAN-CFD architecture introduces the Domain-Group KAN Detector (DG-KD) to synergize locality with fitting capability. The DG-KD employs specific DG-Layers to model new tasks while preserving old ones without interference. To facilitate input space separation for the detector, the FS-KDCP module projects stored previous features into the current space via KAN projection. This effectively compensates for feature semantic drift and separates feature distributions by feature replay.

in which Φ_q and $\phi_{q,p}$ are univariate functions corresponding to each variable. The KAN (Liu et al. 2024) architecture introduces the concept of a KAN layer. For an input vector with dimensions d_{in}, a KAN layer is defined as a matrix of learnable activation functions with an input dimension of d_{in} and an output dimension of d_{out}:

$$\begin{aligned}\mathrm{KAN}(\boldsymbol{x}) &= \begin{bmatrix}\sum_{i=1}^{d_{in}} \phi_{1,i}(x_i) & \cdots & \sum_{i=1}^{d_{in}} \phi_{d_{out},i}(x_i)\end{bmatrix} \\ &= \Phi \circ \boldsymbol{x}, \text{where} \quad \Phi = \begin{bmatrix}\phi_{1,1}(\cdot) & \cdots & \phi_{1,d_{in}}(\cdot) \\ \vdots & \ddots & \vdots \\ \phi_{d_{out},1}(\cdot) & \cdots & \phi_{d_{out},d_{in}}(\cdot)\end{bmatrix}.\end{aligned} \tag{3}$$

Here, $\boldsymbol{x}$ represents the inputs to the KAN layer, while Φ denotes the one-dimensional univariate function matrix of the KAN layer. The authors parameterize each univariate function using a B-spline (Qin 1998):

$$\phi(x) = \omega_b \mathrm{silu}(x) + \omega_s \sum_i c_i B_i(x). \tag{4}$$

Since spline bases are local, samples $\boldsymbol{x}_{i,t} \in \mathcal{D}_t$ in a feature space where domains $\{\mathcal{D}_i\}_{i=1}^{T}$ are well-separated influence only a few adjacent spline coefficients, without affecting distant coefficients that contain prior knowledge. However, the sparse and recursive computations inherent to B-splines complicate their parallelization on GPUs, posing a challenge for processing high-dimensional inputs. Therefore, we first design a KAN architecture that is capable of processing image data while preserving the necessary locality for domain-incremental learning. Second, we construct a feature space in which the feature distributions of different tasks are distinctly separated without requiring data replay.

Architecture Overview

Our architecture comprises two main components: the Domain-Group KAN Detector (DG-KD) and the data-free replay Feature Separation strategy by KAN Drift Compensation Projection (FS-KDCP). As illustrated in Fig. 2, DG-KD sequentially models information from each domain $\mathcal{D}_t = \{(\mathbf{x}_{i,t}, y_{i,t})\}_{i=1}^{n_t}$ by assigning a unique DG-Layer to the learning domain and subsequently combining these layers. A DG-Layer consists of an RBF matrix with group-wise parameter sharing. If the feature spaces for each domain are sufficiently distinct, the activation function, formed by a combination of locally non-zero RBFs, exhibits locality and local plasticity. The FS-KDCP module is designed to enforce separation between the feature spaces of new and old tasks. We only store extracted features from previous tasks, making the method applicable to scenarios where the original data is inaccessible. However, as the backbone network evolves, these stored legacy features are susceptible to semantic drift and may no longer accurately represent their original distributions. To address this issue, we introduce a KAN projection (KDCP) module trained on the new task. This projection aligns the feature spaces of the current and previous backbones and then maps the stored legacy features into the current feature space to reconstruct their original data distributions. A separation loss is then applied to distinguish between the feature distributions of the new and old tasks. Finally, samples from the new task are processed by DG-KD to facilitate domain modeling and classification.

Domain Group KAN Detector

We propose the Domain-Group KAN detector, an architecture designed for domain-incremental learning. It preserves the essential locality of B-splines while being trainable on image data. To construct activation functions with this de-

sired locality, we select RBFs, as they are inherently local, with a non-zero response only within a finite region. A KAN layer constructed using RBFs can be formulated as:

$$f(\boldsymbol{x}) = \left[\sum_{i=1}^{d_{in}} \phi_{1i}(x_i) \quad \dots \quad \sum_{i=1}^{d_{in}} \phi_{d_{out}i}(x_i)\right], \quad \text{where} \quad \phi_{ij}(x) = \exp\Big(-\tfrac{(x-c_{ij})^2}{2\sigma_{ij}^2}\Big). \tag{5}$$

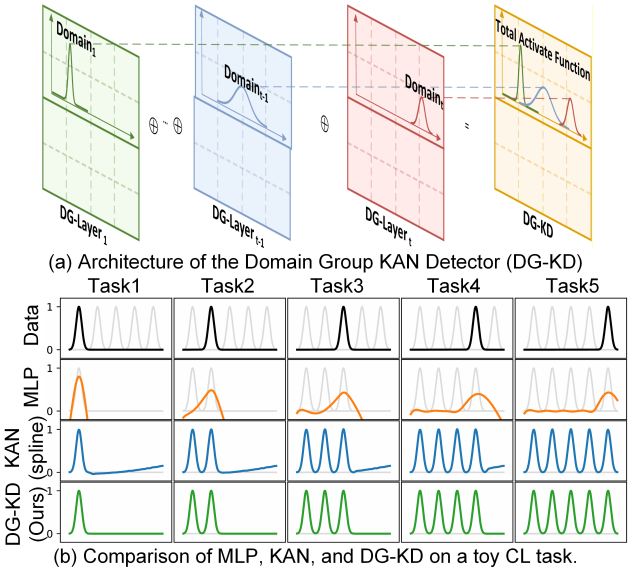
Meanwhile, KAT (Xingyi Yang 2025) identified computational redundancy in KANs, which assign an independent learnable activation function to each input dimension. Building on the premise that RBFs are easily optimized, our work addresses a critical problem of KANs: their representational capacity is often redundant in the feature dimension while being insufficient to model key domain information. Therefore, we introduce the Domain Group Layer (DG-Layer), which is composed of a $d_{in} \times d_{out}$ matrix of RBFs. Inspired by the grouping concept, we partition this matrix into g groups, where all RBFs within each group share the same basis functions to process $d_g = \lfloor \dim(x)/g \rfloor$ dimensions:

$$\begin{aligned} &\text{DG-Layer}_t(\boldsymbol{x}) \\ &= W \times \left[\hat{\phi}_{\lfloor 1/d_g \rfloor, t}(x_1) \quad \dots \quad \hat{\phi}_{\lfloor d_{in}/d_g \rfloor, t}(x_{d_{in}})\right]^\top, \\ &\text{where} \quad W = \begin{bmatrix} \mathrm{w}_{1,1} & \cdots & \mathrm{w}_{1,d_{in}} \\ \vdots & \ddots & \vdots \\ \mathrm{w}_{d_{out},1} & \cdots & \mathrm{w}_{d_{out},d_{in}} \end{bmatrix}. \end{aligned} \tag{6}$$

Here, $\hat{\phi}_{\lfloor i/d_g \rfloor, t}(x_i) = \omega_{i,t} \cdot \phi_{\lfloor i/d_g \rfloor, t}(x_i)$ represents the weighted shared RBF for the $\lfloor i/d_g \rfloor$-th dimensional group within the domain/task t. We define the Domain Group KAN Detector (DG-KD) as the sum of all DG-Layers from task 1 to t, with each layer designed to capture the information corresponding to a particular domain:

$$\begin{aligned} &\text{DG-KD}(\boldsymbol{x}) \\ &= \textstyle\sum_{k=1}^{t} \text{DG-Layer}_k(\boldsymbol{x}) \\ &= \textstyle\sum_{k=1}^{t} W \times \left[\hat{\phi}_{\lfloor 1/d_g \rfloor, k}(x_1) \dots \hat{\phi}_{\lfloor d_{in}/d_g \rfloor, k}(x_{d_{in}})\right]^\top \\ &= W \times \textstyle\sum_{k=1}^{t} \left[\hat{\phi}_{\lfloor 1/d_g \rfloor, k}(x_1) \dots \hat{\phi}_{\lfloor d_{in}/d_g \rfloor, k}(x_{d_{in}})\right]^\top \\ &= W \times \left[\Phi_{\lfloor i/d_g \rfloor}(x_1) \quad \dots \quad \Phi_{\lfloor i/d_g \rfloor}(x_{d_{in}})\right]^\top, \\ &\text{where} \quad \Phi_{\lfloor i/d_g \rfloor}(x_i) = \textstyle\sum_{k=1}^{t} \hat{\phi}_{\lfloor i/d_g \rfloor, k}(x_i). \end{aligned} \tag{7}$$

The DG-KD architecture is intuitively illustrated in Fig. 3(a). Each DG-Layer is designed to model a specific task using its own set of local, weighted RBFs. The final DG-KD is then constructed by summing all individual DG-Layers that comprise the knowledge from all t tasks without interference. We validate the theoretical correctness of DG-KD through a toy continual learning experiment, comparing it against an MLP and a conventional KAN. The task is to sequentially regress five independent, non-overlapping Gaussian peaks (Liu et al. 2024; Lee et al. 2025). As shown in Fig. 3(b), the MLP exhibits severe catastrophic forgetting due to its global activation nature. While the conventional KAN retains prior distributions, it produces undesirable activations in off-domain regions. Our DG-KD, designed for efficient high-dimensional fitting, not only preserves all previously learned distributions but also maintains a zero baseline outside the active domains, thereby achieving a more efficient and robust implementation of local plasticity.



(a) Architecture of the Domain Group KAN Detector (DG-KD)

(b) Comparison of MLP, KAN, and DG-KD on a toy CL task.

Figure 3: Our DG-KD (a) assigns an independent matrix of RBFs, which we term a DG-Layer, to each separated domain. All DG-Layers are then combined to form the final DG-KD. (b) shows a toy continual learning experiment in MLP, Conventional KAN and our DG-KD.

Feature Separation by KAN Drift Compensation

DG-KD successfully preserves locality and local plasticity while accommodating image-based inputs. However, it is still necessary to construct a well-structured feature space in which the feature distributions of different domains are non-overlapping. Such separation ensures that the curves within the DG-KD are not repeatedly modified in the same regions, thereby preventing catastrophic forgetting. Existing methods (Zenke, Poole, and Ganguli 2017; Chaudhry et al. 2019) typically enforce this separation through data replay. Our approach forgoes storing raw data and instead retains only a compact set of representative features from past tasks, which broadens its applicability. After the model completes training on $\mathcal{D}_{t-1}$, we freeze the parameters of this model to serve as a teacher. We then apply the SUR (2025) method to select a sparse and robust set of representative features for $\mathcal{D}_{t-1}$:

$$\{\mathbf{f}_{t-1\to t-1}\} = \text{SUR}(f_\theta^{t-1}(\mathcal{X}_{t-1})), \tag{8}$$

where $\{\mathbf{f}_{t-1\to t-1}\}$ denotes the selected features extracted from data samples $\mathcal{X}_{t-1}$ in the feature space of $\mathcal{D}_{t-1}$, and f_θ^{t-1} serves as the backbone of the model for $\mathcal{D}_{t-1}$. When learning on task $\mathcal{D}_t$, the stored static features gradually deviate from their original positions. Prior research (Yu et al. 2020; Gomez-Villa et al. 2024; Goswami et al. 2024) often formulates this issue as a feature semantic drift problem. To address this problem, we employ a projection module p_{KAN}^t

that consists of a single DG-Layer to compensate for the drift and map $\{\mathbf{f}_{t-1\to t-1}\}$ into the current feature space:

$$\{\mathbf{f}_{t-1\to t}\} = p^t_{\text{KAN}}(\{\mathbf{f}_{t-1\to t-1}\}). \tag{9}$$

Here, $\{\mathbf{f}_{t-1\to t-1}\}$ denotes the selected features extracted from data samples $\boldsymbol{x}_{i,t-1}$ in the feature space of $\mathcal{D}_{t-1}$. We then use feature augmentation (2025) to reconstruct the original distribution. We utilize a supervised contrastive loss to separate features from the previous and current tasks:

$$\mathcal{L}_{\text{SC}} = \sum_{i=1}^{N} \left[-\frac{1}{|P(i)|} \sum_{p\in P(i)} \log \left(\frac{\exp(\mathbf{f}_i \cdot \mathbf{f}_p/\tau)}{\sum_{k\neq i} \exp(\mathbf{f}_i \cdot \mathbf{f}_k/\tau)} \right) \right], \tag{10}$$

where $P(i)$ denotes the set of features $\mathbf{f}_p$ that share the same label as $\mathbf{f}_i$. We assign unique labels to the real and fake samples corresponding to each of the T tasks, resulting in a total of $2T$ labels. Additionally, to constrain the backbone from changing significantly, thereby enabling the projection module to accurately fit its evolution, we introduce a feature-level knowledge distillation loss to regularize the backbone:

$$\mathcal{L}_{\text{KD}} = \frac{1}{N} \sum_{i=1}^{N} \text{MSE}\left(f^{t-1}_{\theta}(x_i), f^{t}_{\theta}(x_i) \right), \tag{11}$$

where N denotes the batch size, and MSE represents the mean squared error loss. Finally, we apply a binary cross-entropy loss $\mathcal{L}_{\text{CLS}}$ for classification. The overall loss is:

$$\mathcal{L}_{\text{Overall}} = \mathcal{L}_{\text{CLS}} + \lambda_1 \mathcal{L}_{\text{SC}} + \lambda_2 \mathcal{L}_{\text{KD}}. \tag{12}$$

KDCP is trained on new task data using an alignment loss:

$$\mathcal{L}_{\text{Align}} = \frac{1}{N} \sum_{i=1}^{N} \text{MSE}\left(p^t_{\text{KAN}}\left(f^{t-1}_{\theta}(x_i) \right), f^{t}_{\theta}(x_i) \right). \tag{13}$$

This loss minimizes the MSE between the features extracted by the current backbone and the mapped features from the previous backbone, thereby aligning the two feature spaces.

Experiments

Datasets

We choose a variety of datasets containing forged faces from multiple domains. FaceForensics++ (FF++) (Rossler et al. 2019) includes 1,000 real videos. These videos are manipulated using four forgery methods: DF, F2F (2016), FS, and NT (2019). The Deepfake Detection (DFD) (Dufour and Gully 2019) dataset provides over 1,000 real and forged face videos. The Deepfake Detection Challenge Preview (DFDC-P) (Dolhansky et al. 2020) contains 5,000 videos utilizing two forgery techniques. Celeb-DF v2 (CDF2) (Li et al. 2020) comprises 590 genuine and 5,639 forged videos. DF40 (Yan et al. 2024) is an extensive dataset featuring 40 distinct forgery methods. Based on these datasets, we establish two evaluation protocols: a dataset incremental protocol and a forgery-type incremental protocol to assess our model's performance. For the dataset incremental protocol, we follow the benchmark proposed by DFIL and select the sequence [FF++, DFDC-P, DFD, CDF2]. For the forgery-type incremental protocol, we strictly adhere to the SUR-LID and select three forgery techniques from DF40: Face-Swapping (FS) with BlendFace, Face-Reenactment (FR) with MCNet, and Entire Face Synthesis (EFS) with StyleGAN3, along with the hybrid forgeries from FF++, collectively constitute the [Hybrid, FR, FS, EFS] testing sequence.

Implementation Details

For data preparation, we utilize standard datasets and data preprocessing methods from DeepFakeBench (Yan et al. 2023) to ensure a fair comparison with existing approaches. For feature extraction, we employ ConvNeXt-B (Liu et al. 2022) as the backbone. For training, we use the Adam (Kingma and Ba 2014) optimizer as our main optimizer, with parameters set to $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate for the main optimizer is set to 2e-4. For the projection optimizer, we use the same Adam optimizer but with a learning rate of 5e-4 to promote faster convergence. The size of our feature memory is 500, which is consistent with existing methods. The batch size is set to 64. The loss hyperparameters are set to $\lambda_1 = 2$, $\lambda_2 = 1$, and $\tau = 0.1$. All experiments are conducted on two Nvidia A6000 GPUs.

We evaluate overall performance using Accuracy (Acc) and Area Under the Curve (AUC). The key Average Forgetting (AF) metric quantifies the model's ability to retain information from previously learned T tasks. It is defined as: $AF = \frac{1}{T-1}\sum_{i=1}^{T-1}(A_{i,i} - A_{T,i})$, where $A_{i,j}$ denotes either Acc or AUC score in T_j after learning T_i. In the comparison experiments, we report the best results from officially published papers for the compared methods, to contrast with our best results. For a fair comparison, we also provide results with aligned backbones in the supplementary material.

Comparison Experiments with SOTA methods

First, we conduct experiments under the dataset-incremental protocol, comparing our method against a comprehensive set of approaches using accuracy (Acc) as the evaluation metric. The compared methods include general continual learning methods such as LWF (Li and Hoiem 2017), as well as continual face forgery detection methods like CoReD (Kim, Tariq, and Woo 2021), DFIL (Pan et al. 2023), DMP (Tian et al. 2024), and SUR-LID (Cheng et al. 2025). The primary experimental results are presented in Table 1. By the end of the training sequence, continual face forgery detection methods exhibit considerable forgetting; CoReD and DFIL register average forgetting rates of 11.42% and 7.01%, respectively. Our method demonstrates superior performance, achieving the highest average Acc of 91.64% and matching the lowest AF of 4.08%. Moreover, while the majority of the compared methods rely on data replay, our feature replay strategy saves 99.48% of the cache size, as confirmed by the efficiency analysis in our supplementary material.

Second, we perform comparative experiments under the forgery-type incremental protocol [Hybrid, FR, FS, EFS]. We include another continual forgery detection method, HDP (Sun et al. 2025) for comparison, using Area Under the Curve (AUC) as the evaluation metric. The experimen-

Method	Dataset	Acc(%)↑				Avg↑	AF↓
		FF++	DFDCP	DFD	CDF2		
LWF* TPAMI' (2017)	FF++	95.52	-	-	-	95.52	-
	DFDCP	87.83	81.57	-	-	84.70	7.69
	DFD	76.16	41.78	96.36	-	71.43	29.58
	CDF2	67.34	67.43	84.05	87.90	76.68	18.21
CoReD MM' (2021)	FF++	95.50	-	-	-	95.50	-
	DFDCP	92.94	87.61	-	-	90.28	2.56
	DFD	86.84	81.07	95.22	-	87.71	7.60
	CDF2	74.08	76.59	93.41	80.78	81.22	11.42
DFIL MM' (2023)	FF++	95.67	-	-	-	95.67	-
	DFDCP	93.15	88.87	-	-	91.01	2.52
	DFD	90.30	85.42	94.67	-	90.03	4.41
	CDF2	86.28	79.53	92.36	83.81	85.49	7.01
DMP MM' (2024)	FF++	95.96	-	-	-	95.96	-
	DFDCP	92.71	89.72	-	-	91.22	3.25
	DFD	92.64	86.09	94.84	-	91.19	3.48
	CDF2	91.61	84.86	91.81	91.67	89.99	**4.08**
SUR-LID CVPR' (2025)	FF++	96.52	-	-	-	96.52	-
	DFDCP	93.35	90.70	-	-	92.03	3.17
	DFD	91.58	87.25	97.50	-	92.11	4.20
	CDF2	90.50	88.08	92.96	92.91	91.11	4.39
KAN-CFD* ***(Ours)***	FF++	97.68	-	-	-	**97.68**	-
	DFDCP	95.90	90.39	-	-	**93.15**	**1.78**
	DFD	93.52	88.83	97.69	-	**93.35**	**2.86**
	CDF2	92.90	87.36	93.26	93.03	**91.64**	**4.08**

Table 1: Experiments on dataset incremental protocol. The best performer is highlighted, while the second-best result is underlined. *denotes methods not requiring data replay.

Method	Type	AUC(%)↑				Avg↑	AF↓
		Hybrid	FR	FS	EFS		
LWF* TPAMI' (2017)	Hybrid	97.00	-	-	-	97.00	-
	FR	88.76	88.45	-	-	88.61	8.24
	FS	84.07	80.99	96.44	-	87.24	10.20
	EFS	78.73	56.73	93.67	92.82	79.32	17.59
CoReD MM' (2021)	Hybrid	96.65	-	-	-	96.65	-
	FR	93.55	79.88	-	-	88.61	3.10
	FS	89.07	79.29	86.05	-	84.80	4.09
	EFS	84.54	64.29	84.17	92.63	81.41	9.86
DFIL MM' (2023)	Hybrid	96.46	-	-	-	96.46	-
	FR	55.74	99.75	-	-	77.75	40.72
	FS	60.71	66.49	99.03	-	75.41	34.51
	EFS	50.83	95.56	70.81	99.96	79.29	26.01
HDP IJCV' (2025)	Hybrid	96.71	-	-	-	96.71	-
	FR	67.41	95.45	-	-	81.43	29.30
	FS	63.00	71.35	95.09	-	76.48	28.91
	EFS	59.89	70.06	89.34	93.73	78.26	22.65
SUR-LID CVPR' (2025)	Hybrid	96.85	-	-	-	96.85	-
	FR	82.91	92.42	-	-	87.66	13.94
	FS	90.50	96.26	97.94	-	**94.90**	**1.26**
	EFS	87.90	96.79	93.56	99.07	94.33	2.99
KAN-CFD* ***(Ours)***	Hybrid	97.02	-	-	-	**97.02**	-
	FR	93.37	92.42	-	-	**92.90**	**3.65**
	FS	90.86	91.23	96.43	-	92.84	3.68
	EFS	90.78	91.81	95.47	99.53	**94.40**	**2.60**

Table 2: Experiments on forgery-type incremental protocol. The best performer is highlighted, while the second result is underlined. *denotes methods not requiring data replay.

tal results are detailed in Table 2. Several compared methods exhibit a severe performance decline early in training. For example, the average forgetting of DFIL after the second task is 40.72%, while SUR-LID and HDP display AFs of 13.94% and 29.30%, respectively. In contrast, our method maintains a consistently low forgetting rate, concluding with a final AF of only 2.60% and achieving the highest AUC of 94.40%. The results from both the dataset-incremental and forgery-type incremental protocols indicate that our method achieves SOTA, surpassing even data replay based methods.

Long-sequence Continual Learning Experiment

To evaluate the capability of our model on long-sequence tasks, we conduct experiments with 10 tasks. These tasks are constructed from different forgery methods in DF40 (2024), covering three major forgery categories: FR, FS, and EFS. For comparison, we select a recent continual forgery detection method, DFIL (2023), and a recent KAN-based continual learning method, KAC (2025). The detailed setup is provided in the supplementary materials. As shown in Fig. 5, our approach achieves the highest average accuracy and the lowest average forgetting rate compared to both methods, demonstrating its effectiveness in long-sequence tasks.

Ablation Study

Ablation Study on the Effect of Loss Functions. We keep the overall framework unchanged and perform the ablation by setting the hyperparameters of each loss term to zero. In Table 3, we find that a model trained using only the classification loss is highly susceptible to catastrophic forgetting, leading to a high AF of 31.35% at the end of the training sequence. Both the $\mathcal{L}_{\text{SC}}$, which serves to separate different domains, and the $\mathcal{L}_{\text{KD}}$, designed to preserve knowledge from previous tasks, effectively mitigate forgetting. Through the combination of these losses, our complete loss function ultimately achieves the best results.

Loss	FF++	DFDC-P		DFD		CDF2	
	AA	AA	AF	AA	AF	AA	AF
$\mathcal{L}_{\text{CLS}}$	97.63	88.97	12.00	80.82	20.43	70.51	31.35
$\mathcal{L}_{\text{CLS}} + \mathcal{L}_{\text{KD}}$	97.04	90.85	5.41	89.33	7.60	85.56	13.44
$\mathcal{L}_{\text{CLS}} + \mathcal{L}_{\text{SC}}$	96.37	91.94	4.26	89.69	8.01	87.87	10.64
$\mathcal{L}_{\text{Overall}}$	**97.68**	**93.15**	**1.78**	**93.35**	**2.86**	**91.64**	**4.08**

Table 3: Ablation study on the effect of loss functions.

Ablation and Visualization of the Feature Separation. We keep the DG-KD unchanged and conduct an ablation experiment to analyze the separation of our feature space. As shown in Table 4, the 'Baseline' (binary classification backbone) suffers from severe forgetting. 'FS w/o KDCP' separates features but neglects drift compensation, leading to degradation as stored features shift over time. 'Ours+Replay' denotes the overall framework but stores raw data from previous tasks, serving as the upper bound of

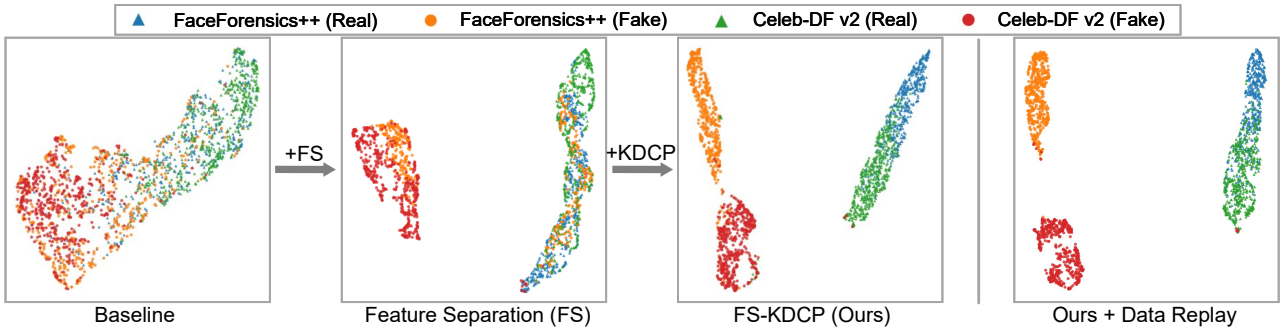


Figure 4: A visual comparison of Feature Separation. The first three plots from the left illustrate the progressive changes in the feature space as Feature Separation (FS) and KAN Drift Compensation Projection (KDCP) are added. The final plot represents the performance upper bound. Notably, our method nearly preserves the structure of the data-replay feature space.

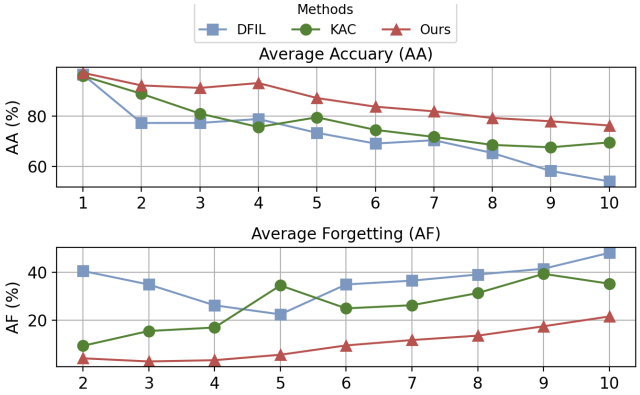


Figure 5: Long-sequence continual learning experiment. The proposed method achieves the best AA and AF.

Module	FF++	DFDC-P		DFD		CDF2	
	AA	AA	AF	AA	AF	AA	AF
Baseline	97.63	88.97	12.00	80.82	20.43	70.51	31.35
FS w/o KDCP	97.42	87.82	12.46	84.99	14.80	83.64	16.04
Ours	**97.68**	**93.15**	**1.78**	**93.35**	**2.86**	**91.64**	**4.08**
Ours+Replay	97.36	93.37	1.07	93.65	2.15	92.63	3.49

Table 4: Ablation study on the feature space constraint.

performance. By introducing KDCP, our method effectively compensates for drift and closely approximates this bound.

To further validate whether KAN-CFD can effectively separate the feature spaces of different tasks, we conduct the visualization experiment shown in Fig. 4. The model sequentially learns two tasks, FF++ and CDF2, and we use UMAP (McInnes, Healy, and Melville 2018) for visualization. We observe that the baseline model suffers from significant feature overlap. Although using only 'Feature Separation (FS)' maintains a relatively clear classification boundary between the two datasets, the FF++ features undergo drift, causing their distribution to significantly overlap with the CDF2 features. This overlap results in some fake samples crossing the decision boundary, leading to poor performance on FF++. In contrast, our method, which incorporates the KAN projection to compensate for drift, effectively preserves the feature space structure of the upper bound without requiring the storage of raw data.

Module	FF++	DFDC-P		DFD		CDF2	
	AA	AA	AF	AA	AF	AA	AF
MLP	96.48	88.38	8.74	78.91	22.76	72.36	28.29
GroupKAN	97.16	88.91	8.33	82.20	17.56	74.42	25.00
KAC	97.29	91.54	4.21	91.57	4.45	86.58	9.77
DG-KD(Ours)	**97.68**	**93.15**	**1.78**	**93.35**	**2.86**	**91.64**	**4.08**

Table 5: Analysis of the Domain Group KAN Detector.

Analysis of the Domain Group KAN Detector. We maintain the FS-KDCP and evaluate the performance of different detectors on continual face forgery detection. The results are presented in Table 5. A standard MLP (1989) employs global activation functions. Consequently, its performance on previous tasks degrades sharply. GroupKAN (2025), which uses rational functions as its activation functions, also suffers from catastrophic forgetting due to its loss of locality. KAC (2025), a method designed for class-incremental learning, achieves satisfactory results due to its robustness in domain-incremental settings. In contrast, our DG-KD, specifically designed for the continual forgery detection task, surpasses KAC and achieves the best results.

Conclusion

In this paper, we propose KAN-CFD, a novel KAN-based method for continual face forgery detection. Our approach overcomes the limitations of KANs by efficiently implementing their locality and adapting it for continual forgery detection task. First, we employ KAN drift compensation projection to map old features into the current feature space, thereby achieving feature space separation without data replay. Second, building upon the separated feature space, we introduce the Domain Group KAN Detector. This detector retains the crucial locality and local plasticity of KAN while enabling the training in image data. Extensive experiments demonstrate that our method achieves SOTA performance.

Acknowledgements

This work was supported in part by Chinese National Natural Science Foundation Projects U23B2054, 62276254, 62176256, Beijing Natural Science Foundation L242092, the Science and Technology Development Fund of Macau Project 0140/2024/AGJ.

References

Aghaei, A. A. 2024. rkan: Rational kolmogorov-arnold networks. *arXiv preprint arXiv:2406.14495*.

Bodner, A. D.; Tepsich, A. S.; Spolski, J. N.; and Pourteau, S. 2024. Convolutional kolmogorov-arnold networks. *arXiv preprint arXiv:2406.13155*.

Buhmann, M. D. 2000. Radial basis functions. *Acta numerica*, 9: 1–38.

Chaudhry, A.; Rohrbach, M.; Elhoseiny, M.; Ajanthan, T.; Dokania, P. K.; Torr, P. H.; and Ranzato, M. 2019. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*.

Cheng, J.; Yan, Z.; Zhang, Y.; Hao, L.; Ai, J.; Zou, Q.; Li, C.; and Wang, Z. 2025. Stacking brick by brick: Aligned feature isolation for incremental face forgery detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 13927–13936.

Dolhansky, B.; Bitton, J.; Pflaum, B.; Lu, J.; Howes, R.; Wang, M.; and Ferrer, C. C. 2020. The deepfake detection challenge (dfdc) dataset. *arXiv preprint arXiv:2006.07397*.

Dufour, N.; and Gully, A. 2019. Contributing data to deepfake detection research. *Google AI Blog*, 1(2): 3.

Gomez-Villa, A.; Goswami, D.; Wang, K.; Andrew, B.; Twardowski, B.; and van de Weijer, J. 2024. Exemplar-free Continual Representation Learning via Learnable Drift Compensation. In *European Conference on Computer Vision*.

Goswami, D.; Soutif-Cormerais, A.; Liu, Y.; Kamath, S.; Twardowski, B.; and van de Weijer, J. 2024. Resurrecting Old Classes with New Data for Exemplar-Free Continual Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hornik, K.; Stinchcombe, M.; and White, H. 1989. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5): 359–366.

Hu, Y.; Liang, Z.; Yang, F.; Hou, Q.; Liu, X.; and Cheng, M.-M. 2025. Kac: Kolmogorov-arnold classifier for continual learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 15297–15307.

Kim, M.; Tariq, S.; and Woo, S. S. 2021. Cored: Generalizing fake media detection with continual representation using distillation. In *Proceedings of the 29th ACM International Conference on Multimedia*, 337–346.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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.

Kolmogorov, A. N. 1961. *On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables*. American Mathematical Society.

Lee, A.; Gomes, H. M.; Zhang, Y.; and Kleijn, W. B. 2025. Kolmogorov-Arnold Networks Still Catastrophically Forget but Differently from MLP. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 18053–18061.

Li, Y.; Yang, X.; Sun, P.; Qi, H.; and Lyu, S. 2020. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3207–3216.

Li, Z. 2024. Kolmogorov-arnold networks are radial basis function networks. *arXiv preprint arXiv:2405.06721*.

Li, Z.; and Hoiem, D. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12): 2935–2947.

Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; and Xie, S. 2022. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11976–11986.

Liu, Z.; Wang, Y.; Vaidya, S.; Ruehle, F.; Halverson, J.; Soljačić, M.; Hou, T. Y.; and Tegmark, M. 2024. Kan: Kolmogorov-arnold networks. *arXiv preprint arXiv:2404.19756*.

McInnes, L.; Healy, J.; and Melville, J. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Nguyen, D.; Mejri, N.; Singh, I. P.; Kuleshova, P.; Astrid, M.; Kacem, A.; Ghorbel, E.; and Aouada, D. 2024. LAA-Net: Localized Artifact Attention Network for Quality-Agnostic and Generalizable Deepfake Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17395–17405.

Pan, K.; Yin, Y.; Wei, Y.; Lin, F.; Ba, Z.; Liu, Z.; Wang, Z.; Cavallaro, L.; and Ren, K. 2023. Dfil: Deepfake incremental learning by exploiting domain-invariant forgery clues. In *Proceedings of the 31st ACM International Conference on Multimedia*, 8035–8046.

Park, J.-D.; Kim, K.-M.; and Shin, W.-Y. 2024. CF-KAN: Kolmogorov-Arnold network-based collaborative filtering to mitigate catastrophic forgetting in recommender systems. *arXiv preprint arXiv:2409.05878*.

Peng, S.; Zhang, T.; Gao, L.; Zhu, X.; Zhang, H.; Pang, K.; and Lei, Z. 2025. Wmamba: Wavelet-based mamba for face forgery detection. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 4768–4777.

Qin, K. 1998. General matrix representations for B-splines. In *Proceedings Pacific Graphics' 98. Sixth Pacific Conference on Computer Graphics and Applications (Cat. No. 98EX208)*, 37–43. IEEE.

Rossler, A.; Cozzolino, D.; Verdoliva, L.; Riess, C.; Thies, J.; and Nießner, M. 2019. Faceforensics++: Learning to

detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1–11.

Shin, H.; Lee, J. K.; Kim, J.; and Kim, J. 2017. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30.

Sun, K.; Chen, S.; Yao, T.; Sun, X.; Ding, S.; and Ji, R. 2025. Continual face forgery detection via historical distribution preserving. *International Journal of Computer Vision*, 133(3): 1067–1084.

Thies, J.; Zollhöfer, M.; and Nießner, M. 2019. Deferred neural rendering: Image synthesis using neural textures. *ACM Transactions on Graphics (TOG)*, 38(4): 1–12.

Thies, J.; Zollhofer, M.; Stamminger, M.; Theobalt, C.; and Nießner, M. 2016. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2387–2395.

Tian, J.; Yu, C.; Wang, X.; Chen, P.; Xiao, Z.; Han, J.; and Chai, Y. 2024. Dynamic Mixed-Prototype Model for Incremental Deepfake Detection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 8129–8138.

Xingyi Yang, X. W. 2025. Kolmogorov-Arnold Transformer. In *The Thirteenth International Conference on Learning Representations*.

Yan, S.; Xie, J.; and He, X. 2021. Der: Dynamically expandable representation for class incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3014–3023.

Yan, Z.; Yao, T.; Chen, S.; Zhao, Y.; Fu, X.; Zhu, J.; Luo, D.; Yuan, L.; Wang, C.; Ding, S.; et al. 2024. DF40: Toward Next-Generation Deepfake Detection. *arXiv preprint arXiv:2406.13495*.

Yan, Z.; Zhang, Y.; Yuan, X.; Lyu, S.; and Wu, B. 2023. DeepfakeBench: A Comprehensive Benchmark of Deepfake Detection. In Oh, A.; Neumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 4534–4565. Curran Associates, Inc.

Yan, Z.; Zhao, Y.; Chen, S.; Guo, M.; Fu, X.; Yao, T.; Ding, S.; Wu, Y.; and Yuan, L. 2025. Generalizing deepfake video detection with plug-and-play: Video-level blending and spatiotemporal adapter tuning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 12615–12625.

Yu, L.; Twardowski, B.; Liu, X.; Herranz, L.; Wang, K.; Cheng, Y.; Jui, S.; and Weijer, J. v. d. 2020. Semantic drift compensation for class-incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6982–6991.

Zenke, F.; Poole, B.; and Ganguli, S. 2017. Continual learning through synaptic intelligence. In *International conference on machine learning*, 3987–3995. PMLR.