Detecting Cross-domain Deepfake Videos with Contrastive Prototype Learning

Yi Li, Plamen Angelov
School of Computing and Communications, Lancaster University
Lancaster, UK
{y.li154, p.angelov}@lancaster.ac.uk

***Abstract*—Deepfake videos are synthetic media generated using advanced deep learning techniques that manipulate or replace the visual and audio content of an original recording, enabling the creation of highly realistic yet entirely fabricated audiovisual content. The proliferation of such manipulated media poses significant societal risks, including potential misinformation, reputation damage, psychological manipulation, and erosion of trust in digital visual communication. Recent deep learning methods for deepfake detection have emerged, leveraging sophisticated machine learning models that analyze multi-modal cues, including facial inconsistencies, unnatural temporal dynamics, and visual misalignments to distinguish between authentic and synthetic content. However, these state-of-the-art detection approaches often struggle with the domain-shift challenge, where models trained on specific deepfake datasets fail to generalize effectively when confronted with unseen generation techniques or evolving synthesis technologies. To address this critical limitation, we propose a self-supervised contrastive learning framework called CPDD, introducing contrast between features and prototypes of original data to alleviate domain-specific distractions (i.e., deepfake generative models or datasets). We calculate the cosine similarity between two features or prototypes to scale the original distance, clustering the features around closely related prototypes. This process encodes the semantic structures discovered through clustering into the learned embedding space. The extensive experiments show that, compared to various benchmark deepfake detection models and domain generalization techniques, the proposed model achieves state-of-the-art performance on the cross-domain deepfake detection task across a wide range of scenarios.**

***Index Terms*—Deepfake video detection, self-supervised learning, contrastive learning, domain generalization, prototype learning**

I. Introduction

The widespread adoption of smart devices combined with the ubiquity of social media platforms, has driven an exponential surge in online multimedia content. Technological advancements, particularly deep generative models [1]–[3], have further accelerated this trend. However, this proliferation raises significant concerns about the authenticity of such content, as many individuals continue to follow the outdated notion that "seeing is believing," often sharing media without verifying its integrity. Deepfake technology, powered by advanced AI and deep learning (DL) techniques, enables the creation of hyper-realistic fake content by altering media—such as swapping faces in videos, modifying speech in audio, or both. The abundance of online data used to train these models makes detecting such forgeries increasingly complex. Deepfakes undermine trust in visual and auditory evidence due to their highly convincing nature, made possible by sophisticated production tools [4]. The unchecked dissemination of fake media poses serious threats, including eroding trust in journalism, inciting political or religious conflict [5], spreading misinformation, enabling fraud and identity theft, and facilitating harmful activities like revenge porn and celebrity defamation. Consequently, deepfake detection has become a critical area of research, drawing increasing attention from researchers.

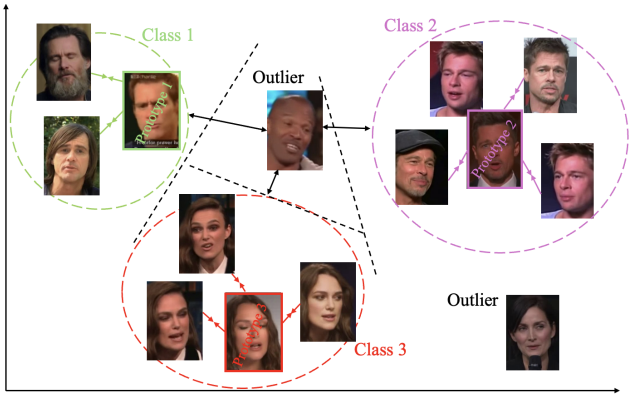


Fig. 1. **Example prototype learning results on Celeb DF [6].** The samples from original videos are clustered, while two outliers, i.e., deepfake videos are kept away from these clusters. Specifically, even though the hair style, presence of a mustache, and apparent age vary across data samples within the green cluster (class 1), the samples are clustered effectively to enable successful classification of the celebrity.

AI-generated videos can generally be categorized into three primary types [4]. (1) Head puppetry [7], or puppet master technique, involves animating a target individual's video as if controlled like a puppet. (2) Face swapping [8] replaces the target person's face with that of a source individual while preserving the target's original facial expressions, creating a realistic yet deceptive video. (3) Lip-syncing [9] modifies the movements of a person's lips to align with specific audio, making it appear as though the individual is speaking words they never actually said. This method focuses on manipulating the lip region to synchronize seamlessly with the target audio, enhancing its deceptive effect.

Recent advances in deep learning have significantly enhanced the effectiveness of deepfake detection, achieving substantial performance improvements [10]. Detection meth-

ods can be broadly classified into two categories: artifact-specific [11] and undirected approaches [12], depending on the data and techniques involved. Artifact-specific methods focus on identifying inconsistencies in deepfake human faces by analyzing features such as edges and optical flow. In contrast, undirected approaches aim to train general-purpose classifiers that analyze entire datasets without relying on specific artifacts, enabling them to learn features autonomously. However, undirected methods have notable limitations.

Most modern deepfake detection techniques [10], [13]–[18] face two primary challenges. First, deepfake detection models' performance can drop significantly when applied to unseen deepfakes created with previously unencountered generative methods or collected under different conditions—a problem known as domain-shift [19], [20]. This challenge occurs because the features or artifacts learned during training may not generalize well to new data distributions, making the model less effective. For instance, differences in resolution, lighting, or compression artifacts can obscure detection-relevant patterns, further reducing accuracy. Addressing domain-shift is crucial, as practical systems must handle the diversity and unpredictability of real-world scenarios without frequent retraining. Second, many detection techniques lack interpretability due to their complex architectures and reliance on high-dimensional feature representations, making their decisions difficult to explain or justify.

To overcome these drawbacks, our contributions are summarized as follows:

- We propose a contrastive prototype learning framework for deepfake video detection (CPDD). The model is pre-trained in a self-supervised manner without the need of any pairs of labelled data.
- Building upon data augmentation (Sec. III-A) and prototype clustering in Sec. III-B, in Sec. III-C, we propose a prototype bank to distinguish individual instances for each prototype from the embedding space. We demonstrate that the instance-wise feature maps capture richer information compared to the prototype-based approach, resulting in performance improvements.
- We provide interpretability to understand the prototype-based classification as the degree to which a human can consistently predict the model's output.
- We demonstrate the efficiency and effectiveness of our proposed methods by comparing them to state-of-the-art deepfake detection models across multi-modal data, i.e., deepfake images and videos.

II. Related Work

A. Deepfake Video Detection

Recently, deep learning techniques [10], [13]–[18] for deepfake video detection have advanced significantly, leveraging a wide range of neural networks to identify subtle spatial-temporal inconsistencies and manipulated features in videos. These techniques can be categorized into temporal sequence analysis methods [15], [17], end-to-end generative adversarial network (GAN) based methods [18], and feature learning [10], [14], [16]. Particularly, GAN-based methods apply adversarial training where detection networks compete against generation networks to detect deepfakes. However, these methods share a common weakness that GANs can be notoriously difficult to train because the generator and the discriminator are constantly competing against others, which can make training unstable and slow. Temporal sequence analysis methods use time-dependent models such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) to detect inconsistencies across video frames and temporal dynamics. For instance, Zheng et al. found that reducing the spatial convolution kernel size to 1, which can improve the detector for extracting the temporal features as well as the generalization capability [17]. In contrast, feature learning-based methods focus on identifying unique audio or visual characteristics that distinguish authentic from manipulated content. For example, Raza et al. extracted audio-visual features from deepfake videos and fed them to multi-label classification head [10]. However, these methods suffer a performance degradation when transferring the trained models to unseen domains (i.e., different data sources, different recording devices, diverse social media platforms, and distinct cultural contexts).

B. Domain Generalization

Recent domain generalization techniques can be divided into several categories based on their motivating intuition.

1) Optimisation Algorithms: Recent optimisation algorithms (e.g., meta-learning and evolving system) solve domain shift problems by minimizing discrepancies between source and target domain distributions [21]. Xie et al. proposed Mutual Information-Based Sequential Autoencoders (MISTS) to adopt information theoretic constraints onto sequential autoencoders to disentangle the dynamic and invariant features, and leverage a domain adaptive classifier to make predictions based on both evolving and invariant information [21].

2) Data Augmentation: In recent data augmentation studies [19], [22], additional training data is synthesized to further improve the model robustness to target domains. Particularly, data augmentation and domain distance minimisation are combined at a high dimensional space in which each axis corresponds to an independent augmentation function.

3) Domain Invariant Features: Assuming that invariant features from source domains perform well in target domains, these features aim to minimize the discrepancy between source domains, facilitating the learning of domain-invariant representations [23]. In adversarial training with generative adversarial networks (GANs), explicit features are generated to deceive the decoder or domain discriminator, ensuring source features become indistinguishable from target features [1].

C. Prototype Learning

Prototype learning is a powerful technique that leverages representative examples, or prototypes, to improve both the performance and interpretability of models [24]. This involves generating latent variables that encapsulate the core features of different classes or clusters within the data, enabling models

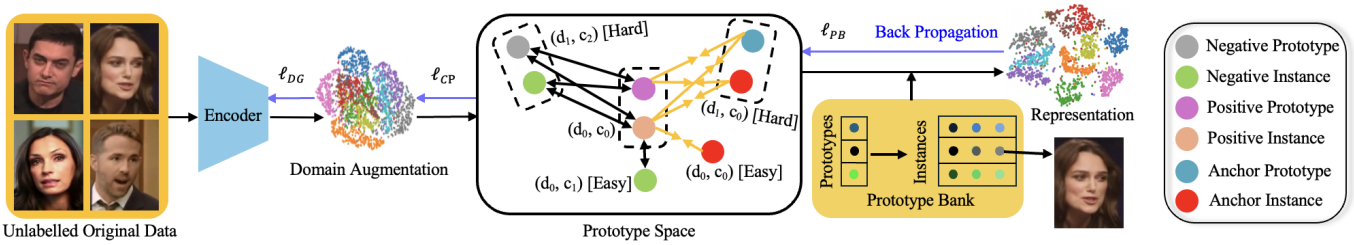


Fig. 2. The proposed contrastive prototype learning framework for cross-domain deepfake video detection.

to generalize more effectively from these prototypes [25]. By focusing on the most distinguishing features of each class, prototype learning simplifies the learning process, enhancing both accuracy and efficiency in decision-making. Prototypes provide concrete examples of what the model deems characteristic of each class, significantly improving the interpretability of deep learning systems [26].

Recent prototype learning advances have yielded notable successes across various domains, including image recognition [27], adversarial attack detection [28]–[30], and generative modeling tasks [31]. These results highlight its ability to foster robust and transparent outcomes, addressing critical challenges in real-world applications. As a result, prototype learning has become an increasingly important tool in advancing both practical implementations and foundational research in artificial intelligence.

III. Proposed Method

We propose a contrastive prototype learning framework to learn robust representation for cross-domain deepfake video detection. The framework is shown in Fig. 2, which includes three contrastive losses as the training objectives.

A. Data Augmentation

As the first training objective of the Transformer-based encoder, we aim to train the parameter θ to transform input data X to feature vectors $V = \{v_1, v_2, ..., v_I\}$, such that V best describes X. Inspired by domain generalization techniques [19], [22], we introduce a contrastive data augmentation loss $\mathcal{L}_{\text{DA}}$ to learn an invariant representation of x_i by minimizing the risk $\sum_i \mathcal{L}(x_i, v_i; \theta)$. Particularly, we adopt a pair of transformations, denoted as t and s, in some set of transformations $\mathcal{T}$ (e.g. flipping, rotation, etc.) to the input x_i, to produce the augmentation $x_i^{o_i}$ and $x_i^{q_i}$. We define the loss as:

$$\mathcal{L}_{\text{DA}} = -\log \frac{\exp\left(f_{DG}\left(x_i^{o_i}\right)^T \cdot f_{DG}\left(x_i^{q_i}\right)/\tau\right)}{\sum_{b=1}^{B} \exp\left(f_{DG}\left(x_b^{o_b}\right)^T \cdot f_{DG}\left(x_i^{q_i}\right)/\tau\right)} \quad (1)$$

where T and B are the transpose symbol and batch size, respectively. We use L2-normalized [32] for all the embeddings in the Eq. 1. While prior data augmentation studies [33] have demonstrated the importance of transformation techniques in self-supervised pre-training, they share a common weakness that lacking the considerations of specific choices of o_i and q_i for frame pairs, instead uniformly sampling transformations from the set $\mathcal{T}$. Therefore, to address this limitation, the proposed data augmentation technique focuses on selecting the optimal transformation algorithm for each sample x_i. This is achieved by choosing transformations that maximize the risk defined by the loss function $\mathcal{L}_{\text{DA}}$:

$$\{o_i, q_i\} = \underset{\{o_i, q_i\} \in \mathcal{T}}{\arg\max} \sum_{i=1}^{n} \mathcal{L}_{\text{DA}}\left(x_i^{o_i}, x_i^{q_i}; \theta, \mathcal{T}\right) \quad (2)$$

In the proposed contrastive data augmentation loss, we prioritize the difference between o_i and q_i for each frame over their absolute values.

B. Contrastive Prototype Learning

As the second training objective of the encoder, we cluster instances based on prototypes. We apply a channel-wise attention to dynamically impose cross-view weights on each single channel to obtain the most discriminative pixels across different views. Specifically, we define i-th pair of the original feature and augmented feature as z_i and z_i' for a single image in a batch, respectively. Then, we obtain the channel-wise integration weight W after a *softmax* normalization:

$$W = \left[\frac{\exp(z_i)}{\exp(z_i) + \exp(z_i')}, \frac{\exp(z_i')}{\exp(z_i) + \exp(z_i')}\right] \quad (3)$$

We obtain the prototype p_i by applying the weights to the features as:

$$p_i = W \times [z_i, z_i'] \quad (4)$$

As the core of this work, the proposed contrastive prototype learning (CPL) can leverage interactions between different instances that share one prototype via a channel-wise attention to enable prototypes focus on encoding the semantic structure of instances into the embedding space. Different from conventional deepfake detection methods, by using the CPL, prototypes eliminate domain-specific features, preserving essential domain-invariant information that facilitates robust generalization across diverse domains. To achieve this, we introduce two contrastive losses, i.e., the feature-prototype contrastive loss $\mathcal{L}_{\text{FP}}$ and the prototype-prototype contrastive loss $\mathcal{L}_{\text{PP}}$ to mine hard pairs for robust representations of feature extractor:

$$\mathcal{L}_{\text{FP}} = \sum_{i=1}^{n} -\log \frac{\exp\left(z_i \cdot p_i^+/\tau\right)}{\sum_{j=0}^{r} \exp\left(z_i \cdot p_j^-/\tau\right)} \quad (5)$$

$$\mathcal{L}_{\text{PP}} = \sum_{i=1}^{n} -\log \frac{\exp\left(p_i \cdot p_j^+/\tau\right)}{\sum_{j=0}^{r} \exp\left(p_i \cdot p_k^-/\tau\right)} \quad (6)$$

where p_i^+ is the positive prototype to the feature z_i and p_j^- includes one postiive embeddings and r negative embeddings for other instances. Moreover, τ is a temperature hyper-parameter. We apply cosine similarity between two features or prototypes, with its reciprocal used to scale the original distance. This means that smaller cosine similarity values result in a larger scaling coefficient applied to the original distance, and vice versa. Additionally, we introduce constraints based on domain information to reduce the number of positive pairs. By leveraging domain labels, this approach minimizes redundancy from easily identifiable positive pairs, guiding the model's learning more effectively. At the same time, the cosine similarity encourages the model to focus on extremely hard positive pairs. Consequently, the combined contrastive prototype loss function maximizes the utility of knowledge from these challenging pairs, addressing domain shifts more robustly:

$$\mathcal{L}_{\text{CPL}} = \mathcal{L}_{\text{FP}} + \mathcal{L}_{\text{PP}} \tag{7}$$

In the proposed contrastive losses, $\mathcal{L}_{\text{FP}}$ pulls original features closer to prototypes from the same category but different domains to resolve domain confusion caused by hard $positive$ pairs. Moreover, $\mathcal{L}_{\text{FP}}$ between prototypes obviates domain confusion to resolve hard $negative$ pairs and form more domain-invariant pairs.

The conventional channel-wise attention generates attention maps that guide the model to focus on specific channels of feature. However, in this work, we apply a channel-wise attention to generate a cross-instance weights that can integrate diverse instances related to the same prototype to effectively cluster the instances. To achieve this, the channel-wise attention focuses on the most semantic pixel across different instances to obtain domain-invariant information.

C. Prototype Bank

As the third training objective of the encoder, we establish a connection between prototype and instance features to facilitate instance clustering. To achieve this, inspired by the recent success of memory banks [34], we propose a prototype bank to cluster instances sharing a common prototype. Specifically, we initialize K independent prototype banks to enhance instance discrimination across different clusters. Much like a memory bank, the prototype bank facilitates contrastive learning by leveraging a large pool of data, enabling the model to acquire more robust and generalizable representations. We assume a contrastive set J_i for the t-th bank A_t as:

$$J_i = \{z_i' \mid z_i' \in A_t \forall t \in [1, C]\} \tag{8}$$

where z_i' is the estimated representation of x_i. Particularly, our prototype memory is set up with size $M \times B \times D$ for each training batch with B samples, D dimensions of pixel embeddings and M prototypes. We use an average pooling on all the embeddings of pixels labeled as p^m prototype in the b-th batch to obtain the a D-dimensional feature vector in the prototype memory, denoted as the (p^m, b)-th element in the memory. Then, to update the prototype bank, we enqueue each instance to the nearest prototype and add the new one in each backpropogation cycle:

$$\mathcal{L}_{\text{PB}} = \frac{\exp(\cos(m_i, z_i) \cdot \cos\left(m_i, p_i^m/\phi\right))}{\sum_{z' \in A_t} \sum_{j=0}^{r} \exp(\cos(m_i, z_j') \cdot \cos\left(m_i, p_j^m/\phi\right)) \cdot J_i} \tag{9}$$

where m_i is the m-th momentum feature and $\cos(\cdot, \cdot)$ is the cosine similarity between a pair of representations. Moreover, ϕ denotes the concentration level of $\mathcal{L}_{\text{PB}}$ and is estimated as:

$$\phi = \frac{\sum_{i=1}^{n} \|p^m - z_i^m\|_2}{n \log n} \tag{10}$$

Therefore, $\mathcal{L}_{\text{PB}}$ helps discriminate representations associated to the same prototype bank. To uncover underlying concepts with distinct visual characteristics, we infer decision boundaries by minimizing visual redundancy among clusters. This is achieved by maximizing the visual similarity of samples within the same cluster while minimizing the similarity between clusters. Specifically, since representations of samples with different pseudo labels are stored independently in the prototype bank, these representations serve as anchors to describe and define their corresponding clusters effectively. The overall cost-function used to train the encoder is now a combination of the above loss terms with hyper-parameters λ_1 and λ_2:

$$\mathcal{L} = \mathcal{L}_{\text{DA}} + \lambda_1 \cdot \mathcal{L}_{\text{CPL}} + \lambda_2 \cdot \mathcal{L}_{\text{PB}} \tag{11}$$

We use a Vision Transformer-Base (ViT-B) as the encoder backbone with different additional layers for contrastive losses. Specifically, we use two Conv1D layers with ReLU for $\mathcal{L}_{\text{CPL}}$. The prototype bank consists of two Conv1D layers with ReLU and faiss [35] for efficient instance clustering. For each prototype, we set the maximum size of the instance queue as 10. It is highlighted that these additional layers are discarded after the pre-training, therefore they do not introduce extra computational cost in deployment. Furthermore, channel attention is usually made up of additional blocks that consume extra additional parameters, while our design is a purely computational module without any additional parameters.

IV. Experiments

A. Data and Deepfakes

We extensively conduct experiments over three public datasets [6], [36], [37] to evaluate the performance of CPDD over deepfake videos.

1) CIFAKE: In the CIFAKE dataset [36], deepfake data includes non-human classes such as airplanes, frogs, and cats. There are 60,000 pairs of real images collected from CIFAR-10 [38] and synthetically-generated images by using a fine-tuned Stable Diffusion Model [3], [39].

2) Celeb-DF: Different from CIFAKE [36], Celeb-DF includes 590 real videos and 5,639 deepfake videos. These real videos featuring 59 celebrities of diverse genders, ages, and ethnic groups are collected from publicly available sources such as YouTube. These deepfake videos are generated by improved synthesis methods, including temporal flickering, inaccurate face masks, and color mismatch, which leads to a significantly enhanced overall visual quality.

3) FaceForensics++: FaceForensics++ [37] comprises 977 videos sourced from YouTube and 1,000 original video sequences featuring unobstructed, easily trackable faces. These sequences are further augmented with manipulated versions created using four techniques: Deepfakes, Face2Face, FaceSwap, and NeuralTextures. Additionally, the dataset includes all the Deepfakes models used in the generation process. Similarly, the DeepFakeDetection dataset features over 363 original video sequences recorded with 28 paid actors across 16 unique scenes, along with more than 3,000 manipulated videos generated using the DeepFakes method.

B. Implementation Details

In the pre-training, we use a ViT-B as the encoder backbone. Due to the limitation of industrial-level computational costs to conventional ViT-based video processing methods [40], in this work, we use frame aggregation [41] to efficiently lower the computational cost with linear complexity in time, at the expense of not considering inter-frame correlation. We pre-train the model using the AdamW optimizer with a momentum of 0.9, an accumulated batch size of 512, and a learning rate of 0.0002. We pre-train for 400 epochs for the encoder. In terms of hyper-parameters, we set $\tau = 0.1$, $\lambda_1 = 1$ and $\lambda_2 = 1$. The supportive experiments for backbone setting are presented in Section IV-G. All experiments are run on Tesla V100 GPUs.

We pre-train the model on 1000 hours of video segments randomly selected from AVSpeech [42]. The dataset is derived from publicly available instructional YouTube videos, including talks, lectures, and how-to tutorials. Each video clip features a single speaking individual as the sole visible face and audible voice in the soundtrack. We sample each video clip at 1 frame per second (FPS).

As aforementioned, after the pre-training, we discard the prototype bank and use the average-pooled top-layer outputs for downstream tasks. We use standard video augmentations [43]. During the fine-tuning stage, we fine-tune the model using different datasets in various experiments, as described in the following subsections.

C. Competitor Models

Our model is evaluated and compared to state-of-the-art competitor models. We reproduce five state-of-the-art deepfake detection techniques [10], [15]–[18], utilizing the best-reported implementations available in the literature. For example, Wang et al. [16] achieve superior results over real image denoising network (RIDNet) [44]. Therefore, we reproduce it with our dataset to serve as a competitor in our comparison experiments. Secondly, we fine-tune two pre-trained domain generalization models [21], [45].

D. Transferring to Unseen Datasets

We first evaluate the transferability of our model in unseen datasets. To update the learned prototype clusters for the downstream task (i.e., deepfake detection), we fine-tune the model using 10% of labeled training videos from one dataset in [6], [36], [37] and evaluate its performance on both real and deepfake videos from the other two datasets. We present the results in Table I.

TABLE I
CROSS-DATASET DEEPFAKE VIDEO DETECTION COMPARISON ON CIFAKE [36], CELEB-DF (Celeb), AND FACEFORENSICS++ (FF) [37]. FT REFERS TO THE FINE-TUNING DATASET.

	FT: Celeb		FT: CIFAR		FT: FF	
Test in →	CIFAKE	FF	Celeb	FF	CIFAKE	Celeb
MISTS [21]	76.1	88.4	79.6	76.0	73.3	84.2
D^3G [45]	76.3	89.3	80.2	76.4	73.8	85.0
DDGAN [18]	61.0	72.5	65.7	64.1	62.6	67.9
NoiseDF [16]	65.8	77.1	69.6	67.2	64.7	70.1
MMtrace [10]	71.3	84.2	73.7	69.8	69.2	78.5
FTCN [17]	76.7	89.0	80.5	76.8	74.3	86.9
ISTVT [15]	77.1	89.9	81.1	77.0	72.8	84.1
CPDD	**78.2**	**91.5**	**82.9**	**78.8**	**74.3**	**87.1**

From Table I, it can be observed that: (1) In all the evaluated models, our model obtains the state-of-the-art deepfake video detection accuracy over all unseen data. This can be attributed to the benefits of learned domain-invariant prototypes and the learned embedding space which encodes the semantic structure of data by prototypical contrastive learning. (2) The detection accuracy is relatively low when CIFAR is used for training or CIFAKE is used for testing. This is because the classes in these classes entirely differ from other datasets. However, our model demonstrates robust performance in addressing this challenge. We also provide the confusion matrices of D^3G, FTCN [17], and our model to show the detailed cross-domain detection accuracy (FaceForensics++ → Celeb-DF) in Fig. 3.

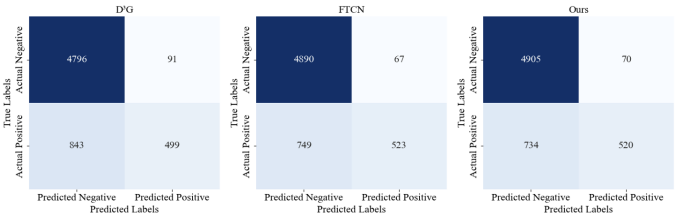


Fig. 3. Confusion matrices on Celeb DF [6].

The proposed CPDD framework learns prototypes from data samples to enhance interpretability. By calculating the cosine similarity between an input image and all identified prototypes, we generate rule-based linguistic representations for each specific sample, enabling a clear explanation of the model's behavior.

E. Transferring to Unseen Deepfake Generation Models

As aforementioned, a wide range of deep generative models [2], [3], [46] are employed in deepfake generation. To assess the robustness of our learned representation, we evaluate its performance across these generative models. Notably, when evaluating on a specific deepfake dataset, we incorporate videos from a mixture of other datasets during the fine-tuning of our model or the reproduction of competitor models. The results are presented in Table II.

TABLE II
DEEPFAKE VIDEO DETECTION COMPARISON OVER STYLEGAN (SG) [2], LATENT FLOW DIFFUSION (LFD) [46], STABLE DIFFUSION (SD) [3] ON CIFAKE [36], CELEB-DF (Celeb) [6], AND FACEFORENSICS++ (FF) [37].

	Celeb			CIFAR			FF		
	SG	LFD	SD	SG	LFD	SD	SG	LFD	SD
MISTS [21]	81.2	78.6	73.5	75.4	73.1	70.2	76.6	74.4	74.7
D^3G [45]	83.0	78.9	72.8	75.5	74.2	69.1	77.4	75.2	74.2
DDGAN [18]	67.5	65.2	58.7	61.0	59.3	56.2	66.6	65.0	61.4
NoiseDF [16]	70.6	68.8	63.1	64.7	63.5	60.0	69.9	66.9	63.7
MMtrace [10]	77.2	73.4	68.6	70.3	69.2	66.1	73.7	72.5	69.0
FTCN [17]	81.7	78.1	71.9	74.8	73.9	69.7	77.3	74.7	73.2
ISTVT [15]	83.8	79.6	74.3	74.9	75.0	70.4	79.1	76.8	74.0
CPDD	**86.8**	**84.7**	**80.1**	**78.0**	**78.8**	**75.6**	**83.4**	**79.2**	**76.5**

Table II shows a robust performance of our model over different deepfake geeration models and datasets. Compared to ISTVT [15], our model has a significant improvement (i.e., 5.2%).

F. Transferring to Unseen Deepfake Techniques

Since real-world applications are often applied to unseen deepfakes created with previously unencountered generative techniques, we then evaluate the transferability of our model in these techniques (head puppetry [7], face swapping [8] and lip-syncing [9]). The results are presented in Table III.

TABLE III
DEEPFAKE VIDEO DETECTION COMPARISON OVER HEAD PUPPETRY (HG) [7], FACE SWAPPING (FS) [8] AND LIP-SYNCING (LS) [9] ON CELEB-DF [6] AND FACEFORENSICS++ [37].

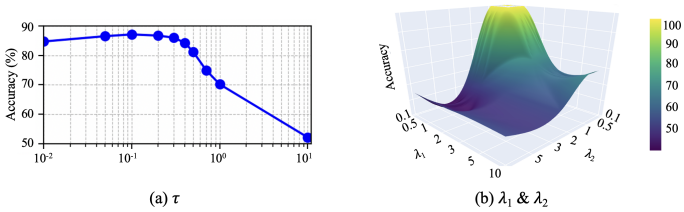
	Celeb-DF			FaceForensics++		
	HG	FS	LS	HG	FS	LS
MISTS [21]	81.5	84.0	80.6	89.5	91.9	89.0
D^3G [45]	81.9	85.1	81.4	90.2	92.0	89.8
DDGAN [18]	66.3	68.3	64.8	81.5	82.2	81.0
NoiseDF [16]	67.2	69.8	65.7	81.9	83.5	81.3
MMtrace [10]	77.5	78.1	77.3	85.9	86.5	86.0
FTCN [17]	82.9	86.7	81.3	86.6	87.4	86.8
ISTVT [15]	80.7	84.4	79.2	89.3	90.0	89.5
CPDD	**88.0**	**89.5**	**86.6**	**94.9**	**95.6**	**95.1**

From Table III, it can be observed that: (1) Our model outperforms competitor models across various deepfake generation techniques and datasets. (2) Self-supervised deepfake detection models [21], [45] outperform supervised models, highlighting the robustness of the self-supervised approach and further supporting our motivation.

G. Hyper-parameters

We then examine the deepfake detection accuracy of our model against hyper-parameters on Celeb-DF. The results are presented in Fig. 4.

As Fig. 4(a) shows, detection accuracy starts to increase with $\tau = 0.01$ and reaches its peak around $\tau = 0.1$. Fig. 4(b) presents detection accuracy against λ_1 and λ_2. There is no significant accuracy drop even when the importance of loss terms is significantly weighted, such as by as much as tenfold that of $\mathcal{L}_{\text{PM}}$ ($\lambda_1, \lambda_2 = 1$). This demonstrates that the features derived from each loss terms contribute positively to the learning process.



(a) τ (b) λ_1 & λ_2

Fig. 4. Ablation study for (a) hyper-parameter τ and (b) $\lambda_1 \& \lambda_2$.

H. Ablation Study

In this section, we evaluate the effectiveness of each proposed contrastive loss in each training objective and compare them to recent contrastive losses [47], [48]. Since each of the three losses proposed in this paper depends on the preceding one, conducting an ablation study without the previous loss while retaining the current loss is not feasible. We train the models on FaceForensics++ and test on Celeb-DF. The ablation study results are presented in Table IV.

TABLE IV
ABLATION STUDY OF THREE CONTRASTIVE LOSSES IN THE PROPOSED METHOD.

Method	Accuracy (%)
ViT (baseline)	57.0
Baseline	
+ $\mathcal{L}_{\text{InfoNCE}}$ [47]	54.0
+ $\mathcal{L}_{\text{UCL}}$ [48]	56.8
+ $\mathcal{L}_{\text{SupCon}}$ [49]	59.3
+ $\mathcal{L}_{\text{DA}}$	60.4
Baseline+$\mathcal{L}_{\text{DA}}$	
+ $\mathcal{L}_{\text{CPL}}$	79.6
Baseline+$\mathcal{L}_{\text{DA}}$+$\mathcal{L}_{\text{CPL}}$	
+ $\mathcal{L}_{\text{PB}}$	**87.1**

Table IV shows that: (1) Each of our proposed losses contributes significantly to the improvement in performance. Specifically, $\boldsymbol{\mathcal{L}_{\text{CPL}}}$ contributes the most improvement (e.g., 19.2%). (2) Our contrastive losses outperform the conventional losses [47]–[49].

I. Visualizations

As qualitative analysis, Fig. 5 presents the t-distributed stochastic neighbour embedding (t-SNE) visualisation of our model trained with different losses. Compared to the representation learned by $\mathcal{L}_{\text{DA}}$, the representation learned by two losses ($\mathcal{L}_{\text{DA}}$ and $\mathcal{L}_{\text{CPL}}$) forms more separated clusters, which also suggests representation of lower entropy. In Fig. 5(e), it can be observed that the feature embeddings within the brown and red classes are not separable. However, when the prototype bank is added in Fig. 5(f), individual instances become separated. This demonstrates that the proposed methods can learn discriminative feature representations that generalize well for deepfake detection across various scenarios.

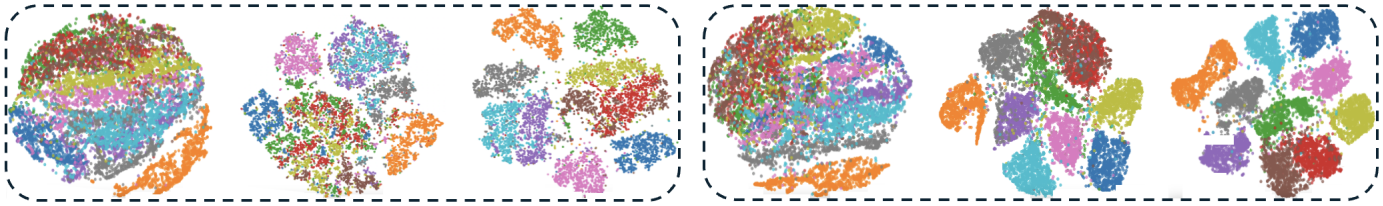


Fig. 5. t-SNE feature visualizations of the model with different losses. First three sub-figures refer to (a) $\mathcal{L}_{\text{DA}}$ (b) $\mathcal{L}_{\text{DA}}+\mathcal{L}_{\text{CPL}}$ (c) $\mathcal{L}_{\text{DA}}+\mathcal{L}_{\text{CPL}}+\mathcal{L}_{\text{PB}}$ of the top 10 classes. The last three sub-figures refer to (d) $\mathcal{L}_{\text{DA}}$ (e) $\mathcal{L}_{\text{DA}}+\mathcal{L}_{\text{CPL}}$ (f) $\mathcal{L}_{\text{DA}}+\mathcal{L}_{\text{CPL}}+\mathcal{L}_{\text{PB}}$ of the next 10 classes.

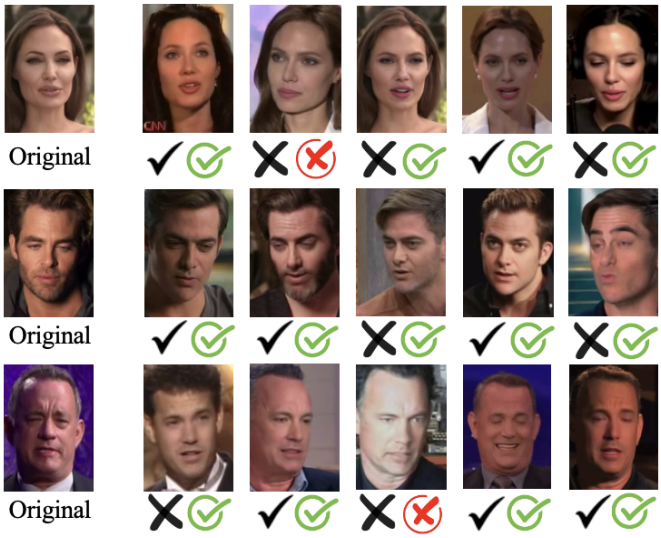


Fig. 6. **Example prototype learning results on Celeb DF [6].** ✓ and ✗ refer to correct and wrong predictions of ISTVT [15], while green and red marks refer to correct and wrong predictions of our model, respectively.

Secondly, we present some qualitative result in Fig. 6 to show the effectiveness of our model. The detection results of our model and ISTVT [15] are denoted by green/red and black, respectively.

As qualitative analysis, Fig. 6 presents the deepfake detection results by using ISTVT and our model. We observe the following: (1) Both ISTVT and our model successfully detect most generated celebrity videos; (2) Our model outperforms ISTVT in detecting certain generated videos because prototypes capture the most salient and generalizable characteristics of each class, enabling the model to distinguish between real and deepfake videos effectively; (3) Our model fails to detect the third generated video of the third celebrity. This failure may be attributed to overexposure, which prevents the prototypes from fully capturing the diversity within the classes.

V. Conclusion

We propose a self-supervised contrastive learning framework for cross-domain deepfake detection. Different from conventional deepfake detection techniques, our approach introduces contrast between features and prototypes of original data to mitigate domain-specific distractions. Evaluations on deepfake video datasets demonstrate the robust performance of the proposed method on cross-domain data, including unseen deepfake datasets and generative techniques. Furthermore, as the most representative samples within classes, prototypes enhance the explainability and interpretability of the network's predictions.

Acknowledgment

This work is supported by ELSA – European Lighthouse on Secure and Safe AI funded by the European Union under grant agreement No. 101070617.

References

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Conference on Neural Information Processing Systems (NeurIPS)*, 2014.

[2] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[3] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," *arXiv preprint arXiv:2112.10752*, 2021.

[4] T. Zhang, "Deepfake generation and detection, a survey," *Multimedia Tools Appl.*, vol. 81, no. 5, p. 6259–6276, 2022.

[5] S. Burgess, "Ukraine war: Deepfake video of Zelenskyy telling Ukrainians to 'lay down arms' debunked," *Sky News*, 2023.

[6] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-DF: A large-scale challenging dataset for deepfake forensics," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[7] H. Lee, C. Lee, K. Farhat, L. Qiu, S. Geluso, A. Kim, and O. Etzioni, "The tug-of-war between deepfake generation and detection," *Proceedings of the International Conference on Machine Learning (ICML)*, 2022.

[8] G. Stanishevskii, J. Steczkiewicz, T. Szczepanik, S. Tadeja, J. Tabor, and P. Spurek, "ImplicitDeepfake: Plausible face-swapping through implicit deepfake generation using nerf and gaussian splatting," *arXiv preprint arXiv:2402.06390*, 2024.

[9] S. K. Datta, S. Jia, and S. Lyu, "Exposing lip-syncing deepfakes from mouth inconsistencies," *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, 2024.

[10] M. A. Raza and K. Malik, "Multimodaltrace: Deepfake detection using audiovisual representation learning," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[11] E. Josephs, C. Fosco, and A. Oliva, "Artifact magnification on deepfake videos increases human detection and subjective confidence," *Journal of Vision*, vol. 23, p. 5327, 2023.

[12] A. Khormali and J.-S. Yuan, "Self-supervised graph Transformer for deepfake detection," *arXiv preprint arXiv:2307.15019*, 2023.

[13] A. L. Pellcier, Y. Li, and P. Angelov, "PUDD: Towards robust multi-modal prototype-based deepfake detection," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[14] A. Chintha, A. Rao, S. Sohrawardi, K. Bhatt, M. Wright, and R. Ptucha, "Leveraging edges and optical flow on faces for deepfake detection," *Proceedings of IEEE International Joint Conference on Biometrics (IJCB)*, 2020.

[15] C. Zhao, C. Wang, G. Hu, H. Chen, C. Liu, and J. Tang, "ISTVT: Interpretable spatial-temporal video Transformer for deepfake detection," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 1335 – 1348, 2023.

[16] T. Wang and K. Chow, "Noise based deepfake detection via multi-head relative-interaction," *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*, 2023.

[17] Y. Zheng, J. Bao, D. Chen, M. Zeng, and F. Wen, "Exploring temporal coherence for more general video face forgery detection," *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[18] P. Sharma, M. Kumar, and H. K. Sharma, "A GAN-based model of deepfake detection in social media," *Procedia Computer Science*, vol. 218, pp. 2153–2162, 2023.

[19] H. S. Le, R. Akmeliawati, and G. Carneiro, "Domain generalisation with domain augmented supervised contrastive learning," *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*, 2020.

[20] Y. Li, Y. Sun, K. Horoshenkov, and S. M. Naqvi, "Domain adaptation and autoencoder based unsupervised speech enhancement," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 1, pp. 43 – 52, 2021.

[21] B. Xie, Y. Chen, J. Wang, K. Zhou, B. Han, W. Meng, and J. Cheng, "Enhancing evolving domain generalization through dynamic latent representations," *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*, 2024.

[22] Z. Wang and Z. Wang, "A domain transfer based data augmentation method for automated respiratory classification," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

[23] Q. Lang, L. Zhang, W. Shi, W. Chen, and S. Pu, "Exploring implicit domain-invariant features for domain adaptive object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[24] Y. Li, P. Angelov, and N. Suri, "Self-supervised representation learning for adversarial attack detection," *Proceedings of European Conference on Computer Vision (ECCV)*, 2024.

[25] G. Li, V. Jampani, L. Sevilla-Lara, D. Sun, J. Kim, and J. Kim, "Adaptive prototype learning and allocation for few-shot segmentation," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[26] P. Angelov and E. Soares, "Towards explainable deep neural networks (xDNN)," *Neural Networks*, vol. 130, p. 185–194, 2020.

[27] M. G. L. Gallee, M. Beer, "Interpretable medical image classification using prototype learning and privileged information," *Proceedings of Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2023.

[28] Y. Li, P. Angelov, and N. Suri, "Robust self-supervised learning for adversarial attack detection," *Proceedings of Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[29] A. L. Pellcier, K. Giatgong, Y. Li, N. Suri, and P. Angelov, "UNICAD: A unified approach for attack detection, noise reduction and novel class identification," *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2024.

[30] Y. Li, P. Angelov, and N. Suri, "Adversarial attack detection via fuzzy predictions," *IEEE Transactions on Fuzzy Systems*, vol. 32, no. 12, pp. 7015 – 7024, 2024.

[31] J. He, T. Berg-Kirkpatrick, and G. Neubig, "Learning sparse prototypes for text generation," *Proceedings of Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[32] M. Yang, Z. Meng, and I. King, "FeatureNorm: L2 feature normalization for dynamic graph embedding," *Proceedings of IEEE International Conference on Data Mining (ICDM)*, 2020.

[33] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *Proceedings of International Conference on Machine Learning (ICML)*, 2020.

[34] S. Yokoo, "Contrastive learning with large memory bank and negative embedding subtraction for accurate copy detection," *arXiv preprint arXiv:2112.04323*, 2021.

[35] J. Johnson, M. Douze, and H. Jegou, "Billion-scale similarity search with GPUs," *IEEE Transactions on Big Data*, vol. 7, p. 4249–4260, 2021.

[36] J. J. Bird and A. Lcifi, "CIFAKE: Image classification and explainable identification of AI-generated synthetic images," *IEEE Access*, p. 99, 2024.

[37] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics++: Learning to detect manipulated facial images," *Proceedings of International Conference on Computer Vision (ICCV)*, 2019.

[38] A. Krizhevsky, "Learning multiple layers of features from tiny images," *Master's thesis*, 2009.

[39] Y. Li, Y. Sun, and P. Angelov, "Complex-cycle-consistent diffusion model for monaural speech enhancement," *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*, 2025.

[40] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Luci, and C. Schmid, "ViViT: a video vision Transformer," *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[41] J. Lee, E. Kim, S. Lee, J. Lee, and S. Yoon, "Frame-to-frame aggregation of active regions in web videos for weakly supervised semantic segmentation," *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[42] A. Ephrat, I. Mosseri, O. Lang, T. Dekel, K. Wilson, A. Hassidim, W. T. Freeman, and M. Rubinstein, "Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation," *Proceedings of ACM SIGGRAPH*, 2018.

[43] C. Wei, H. Fan, S. Xie, C. Wu, A. L. Yuille, and C. Feichtenhofer, "Masked feature prediction for self-supervised visual pre-training," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[44] S. Anwar and N. Barnes, "Real image denoising with feature attention," *Proceedings of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[45] H. Yao, X. Yang, X. Pan, S. Liu, P. W. Koh, and C. Finn, "Improving domain generalization with domain relations," *Proceedings of International Conference on Learning Representations (ICLR)*, 2024.

[46] A. C. K, A. V, S. Das, and A. Das, "Latent flow diffusion for deepfake video generation," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[47] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[48] S. Fung, X. Lu, C. Zhang, and C.-T. Li, "DeepfakeUCL: Deepfake detection via unsupervised contrastive learning," *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2021.

[49] Y. Xu, K. Raja, and M. Pedersen, "Supervised contrastive learning for generalizable and explainable deepFakes detection," *Proceedings of IEEE/CVF Winter Conference on Computer Vision (WACV)*, 2022.