FakeRadar: Probing Forgery Outliers to Detect Unknown Deepfake Videos

Zhaolun Li[1] Jichang Li[2*] Yinqi Cai[3] Junye Chen[3]
Xiaonan Luo[1] Guanbin Li[3,2,5] Rushi Lan[1,4*]
[1]Guilin University of Electronic Technology [2]Pengcheng Laboratory [3]Sun Yat-sen University
[4]Guangxi Key Laboratory of Image and Graphic Intelligent Processing
[5]Guangdong Key Laboratory of Big Data Analysis and Processing
zhaolunli@mails.guet.edu.cn, li.jichang@pcl.ac.cn, rslan2016@163.com

This supplementary material provides an extended experimental exploration and in-depth analysis of our proposed model, *FakeRadar*. By thoroughly examining each proposed component, we aim to gain a deeper understanding of its impact and contributions to deepfake detection. Furthermore, we present detailed experimental results and comparative analyses to validate the effectiveness of *FakeRadar* under various experimental settings. **The pseudo-code of FakeRadar is provided in Algorithm** 1. The following sections outline the details of this study.

Effect of Model Variants on Deepfake Detection To systematically evaluate the impact of different components in FakeRadar (our full model, here we refer to as "Proposed"), we design two model variants: FakeRadar (Frozen), which directly employs a pre-trained CLIP [8] ViT-B/16 encoder without fine-tuning; FakeRadar (Supervised), which introduces ST-Adapter [7] and binary classification, with parameter-efficient fine tuning.

As shown in Table 1, the "Frozen" variant performs poorly across all datasets, achieving only 55.2%-60.0% AUC, demonstrating that pre-trained CLIP features alone are insufficient for deepfake detection. The "Supervised" variant, which incorporates ST-Adapter and binary classification, significantly improves performance, achieving 88.2% on CDFv2 [6] and 94.2% on DFD [9]. However, its generalization remains limited when tested on unseen datasets.

The "Proposed" FakeRadar model, trained with Forgery Outlier Probing and Outlier-Guided Tri-Training, consistently outperforms both baselines. It surpasses the "Supervised" variant by 3.5% AUC on CDFv2, 3.7% on DFDCP [2], and 5.8% on DFDC [3], highlighting that its improved performance is attributed to the specialized training strategies rather than solely relying on a strong ViT-B backbone.

*Corresponding Authors.

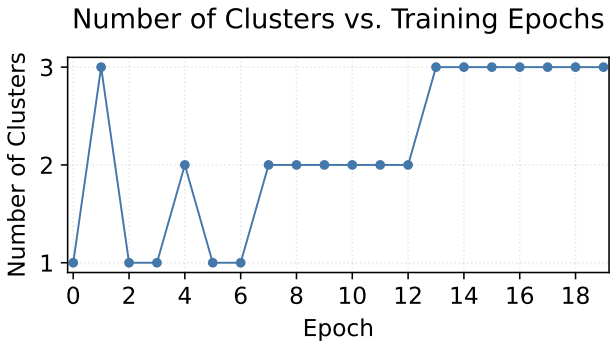

Figure 1. Evolution of the number of clusters during training in Forgery Outlier Probing (FOP). The experiment is conducted training on the FF++(HQ) training set while evaluating on DFDC.

How Shocking Are the Subcluster Fluctuations in Forgery Outlier Probing! In constructing the module of Forgery Outlier Probing, we propose a process of dynamic subcluster modeling, which involves the splitting and merging of subclusters. Using deepfake videos of "DFDC" from the training set as a representative, we analyze the evolution of subcluster numbers across training epochs. Figure 1 illustrates the changes in the number of subclusters constructed during each training epoch, beginning with the initial cluster (where each manipulation type is treated as a separate cluster, and all samples from "DFDC" are initially assigned to the same cluster). As shown, we observe significant fluctuations in the number of subclusters during the early training stages (Epochs 0-8). This indicates that the model is still exploring the feature distribution of the samples and gradually developing its ability to discriminate. After approximately Epoch 10, the number of subclusters stabilizes at three, suggesting that as training progresses, the model's capacity to discriminate the forgeries of clusters improves.

Model Variant	Architecture	Input Type	Training Strategy	AUC (%)				
				FF++	CDF	DFDCP	DFDC	DFD
FakeRadar (Frozen)	ViT-B/16	Frame	No Fine-tuning	55.2	60.0	59.0	55.2	57.4
FakeRadar (Supervised)	ViT-B/16+ST-Adapter	Video	Binary Classification	98.2	88.2	84.8	78.3	94.2
FakeRadar (Proposed)	ViT-B/16+ST-Adapter	Video	FOP+OGTT	**99.1**	**91.7**	**88.5**	**84.1**	**96.2**

Table 1. Comparison of different FakeRadar variants on cross-dataset generalization, evaluated using video-level AUC (%). The "Frozen" variant uses the pre-trained CLIP model without fine-tuning, while the "Supervised" variant integrates ST-Adapter with binary classification. The "Proposed" model further incorporates Forgery Outlier Probing (FOP) and Outlier-Guided Tri-Training (OGTT), leading to significant improvements across all datasets.

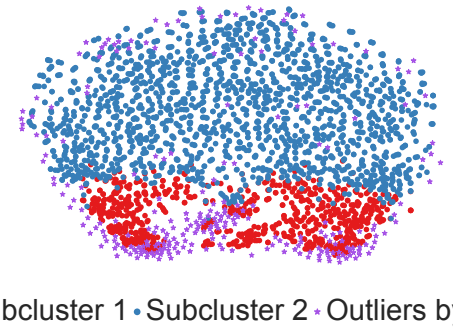

Figure 2. t-SNE [11] visualization of virtual feature-space outliers generated by our Cluster-Conditional Outlier Generation. The known subclusters ("Subcluster 1" and "Subcluster 2") are derived from NeuralTextures and encoded by CLIP's visual encoder [8].

Visualization of Cluster-Conditional Outlier Generation. In this section, we visualize the virtual feature-space outliers synthesized by our Cluster-Conditional Outlier Generation approach using t-SNE [11] (as shown in Figure 2). The visualization involves two known subclusters, labeled as "Subcluster 1" and "Subcluster 2", extracted from NeuralTextures and encoded using CLIP's visual encoder [8]. Note that, as discussed in [4], these synthesized outliers cannot be visualized in pixel space, as they are directly generated within a lower-dimensional feature space. As shown in Figure 2, these virtual outliers clearly reside near the boundaries of the known subclusters, demonstrating that synthesizing unseen forgeries helps the model capture novel forgery traces beyond those of existing real data and known manipulations. This strategy thus significantly enhances FakeRadar's generalization ability in detecting previously unseen deepfake videos.

Correcting Misclassifications in Deepfake Detection with FakeRadar. To validate the overall effectiveness of our proposed FakeRadar, we evaluate the impact of the core components in detection performance of deepfake videos. In our approach, the classifier is designed with three categories: "Real", "Fake" and "Outlier". During evaluating the model, we focus on how the triplet-class classifier

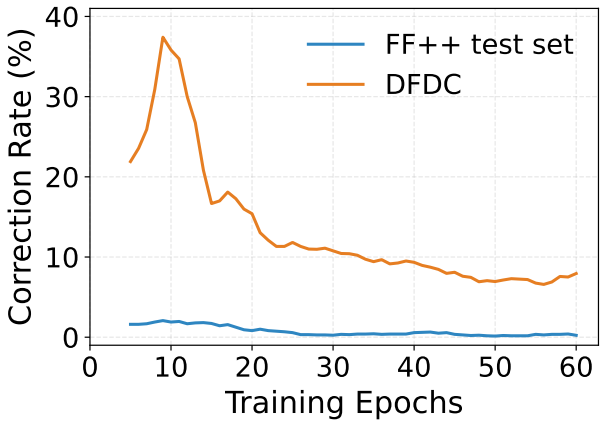

Figure 3. Correction rate of misclassified "Fake" samples over training epochs using the "Fake" + "Outlier" strategy. The curves correspond to the correction rates on the FF++ test set [10] (with the same manipulation patterns as training data) and the DFDC [3] dataset (with different forgery traces from those deepfakes of training data). As training progresses, the correction rates for both datasets initially increase and then decrease, indicating that the model improves in distinguishing deepfake samples over time. Evaluation of Outlier-Conditioned Cross-Entropy Loss on deepfake sample misclassification over training epochs. The model is trained on the FF++(HQ) training set, while evaluating on the FF++ test set and DFDC.

corrects misclassifications of deepfake samples, specifically those misclassified as "Fake", by reclassifying them into a combined category of new "Fake" (denoted as "Fake" + "Outlier"), as illustrated in Figure 3. In the experiment, the model is trained using the FF++(HQ) training set and evaluated on both the FF++ test set (which shares the same manipulation patterns as deepfakes of the training data) and the DFDC dataset (which contains different forgeries from deepfakes in training set). Here, we calculate the proportion of misclassified "Fake" samples that are subsequently corrected by the strategy of "Fake" + "Outlier".

As illustrated, the results show that during training, the correction rates for both the FF++ test set and DFDC initially increase and then decrease. Specifically, around the 10-th epoch, the correction rates reach approximately 5%

Algorithm 1 FakeRadar's Execution Process

Input: Video clips $X = \{x_1, x_2, \ldots, x_n\}$ from real and fake categories
Initialize: Pretrained CLIP's image encoder $\mathbf{M}$, with finetunable ST-Adapter
for each training iteration **do**
// Step 1: Forgery Outlier Probing (FOP)
Dynamic Subcluster Modeling:
  for each video clip $x_i \in X$ **do**
    Extract feature $f(x_i)$ using model $\mathbf{M}$
    Partition features into subclusters using GMM
    Adjust subclusters through dynamical *merging* and *splitting* strategies
  end for
Cluster-Conditional Outlier Generation:
  for each subcluster C_k **do**
    Generate outlier samples near the boundary of subcluster C_k
    Simulate unseen forgeries by generating outliers in the feature space
  end for
// Step 2: Outlier-Guided Tri-Training
  Compute loss for each sample based on its proximity to subcluster centers
  Apply *Outlier-Driven Contrastive Loss* to separate different categories (Real, Fake, Outlier)
  Apply *Outlier-Conditioned Cross-Entropy Loss* to optimize model decision boundaries
end for
// Step 3: Inference
for each test sample x_{test} **do**
  Extract features: $f(x_{test})$
  Classify sample as either "Real", "Fake", or "Outlier" based on decision boundaries
  if Sample is classified as Fake or Outlier **then**
    Output: **Fake**
  else
    Output: **Real**
  end if
end for

and 40%, respectively, before declining to about 3% and 10% at the last. These findings suggest that as the model's performance improves, its ability to discriminate deepfake samples enhances, thereby reducing the need for corrective reclassification.

Additionally, the similarity between the manipulation type in a sample and those in training set significantly influences the outcomes of the "Fake" and "Fake" + "Outlier" categories. For samples with forgery types similar to those in training data, the model assigns a high confidence level through the "Fake" classifier, enabling the binary classifier (Real vs. Fake) to accurately classify the sample. In these cases, the corrective effect of the "Fake" + "Outlier" strategy is limited. However, for samples with forgery types differing from those in training data, the "Fake" classifier assigns a lower confidence level. In such cases, the "Outlier" component substantially enhances prediction confidence, effectively correcting misclassifications.

To sum up, our experimental results demonstrate the following: (1) synthesizing "Outlier" samples to simulate unseen forgeries effectively expands the model's exploration of unknown forgery types and corrects misclassifications within the "Fake" category, thereby validating the effectiveness of our proposed Forgery Outlier Probing; (2) compared to standard binary cross-entropy loss, our proposed Outlier-Conditioned Cross-Entropy Loss offers superior performance by assigning a distinct category to outlier samples, which compels the model to learn a more discriminative decision boundary and prevents misclassification of outliers as real samples; and (3) while the model's inherent discrimination ability improves during training, our approach remains effective, particularly for forgery types that differ from those in training data, where the proposed strategy yields more significant benefits.

Necessity of Dynamic Subcluster Modeling. The goal of dynamic subcluster modeling is to uncover fine-grained patterns within each forgery category. Due to variations in source videos and manipulation techniques, each forgery type typically contains multiple distinct subgroups. Treating these heterogeneous subgroups as a single cluster often obscures low-confidence samples near category boundaries. To address this, we propose the **subclustering network**, which dynamically partitions coarse clusters with high dispersion into more coherent subclusters. Consequently, our outlier generator can sample challenging outliers around cluster boundaries, significantly enhancing the model's generalization capability to unseen forgery types. To validate the effectiveness of our proposed method, we design two model variants for ablation analysis:

(1) A fixed-subcluster variant with $K = 5$, effectively disabling dynamic subcluster adjustment and directly using cluster-conditional outlier generation.
(2) A variant trained without prior knowledge of forgery subtypes, using only two labels (*Real* and *Fake*), where all forgery subtypes are merged into a single *Fake* class.

As shown in Table 2, the fixed-subcluster variant (M-(2)) achieves an average AUC of 87.4%, underperforming our full FakeRadar model (M-(3), 90.1%) by 2.7%. This result highlights that dynamically adapting the number of subclusters (K) enables FakeRadar to better capture subtle intra-

M-(#)	Method	CDFv2	DFDCP	DFDC	DFD	Average
1	FakeRadar (no prior)	91.6	88.1	83.1	95.3	89.5
2	FakeRadar (fixed $K = 5$)	90.0	88.3	82.0	94.1	87.4
3	FakeRadar (Ours)	**91.7**	**88.5**	**84.1**	**96.2**	**90.1**

Table 2. Ablation analysis of FakeRadar with different subcluster modeling strategies. Results reported in AUC (%). The best results are shown in bold.

Method	Saturation	Contrast	Block	Noise	Blur	Average
AltFreezing [12]	**-0.4**	**-0.9**	-8.0	-38.9	**-1.5**	-9.9
StyleFlow [1]	**-0.4**	-3.8	-7.4	-44.6	-2.3	-11.7
FakeRadar (Ours)	-1.3	**-0.9**	**-0.2**	**-27.9**	-7.8	**-7.6**

Table 3. Robustness evaluation of FakeRadar and comparison methods under various perturbations, measured by average AUC drop (%). Best results are highlighted in bold.

category variations, thus significantly improving generalization to unseen forgery types. Moreover, removing fine-grained subtype information only slightly reduces the cross-domain accuracy from 90.1% to 89.5% (M-(3) $\rightarrow$ M-(1)). This indicates that FakeRadar can still effectively generalize without forgery-specific labels, although incorporating detailed subtype information provides additional performance gains.

Model Robustness to Unseen Perturbations. Following prior work [5], we train our model on the FF++ (HQ) dataset and evaluate its robustness across various unseen perturbations at different severity levels. These perturbations include saturation changes, contrast adjustments, block-wise masking, Gaussian noise, and image compression. Robustness is measured by the average drop in AUC scores, as detailed in Table 3. Overall, our proposed FakeRadar demonstrates strong robustness on average across different perturbations compared to other algorithms. Notably, FakeRadar significantly outperforms competing methods under Gaussian noise (average AUC drop: -27.9%) and block-wise masking (average AUC drop: -0.2%), highlighting its superior resilience against challenging corruptions.

References

[1] Jongwook Choi, Taehoon Kim, Yonghyun Jeong, Seungryul Baek, and Jongwon Choi. Exploiting style latent flows for generalizing deepfake video detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1133–1143, 2024. 4

[2] B Dolhansky. The dee pfake detection challenge (dfdc) pre view dataset. *arXiv preprint arXiv:1910.08854*, 2019. 1

[3] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset. *arXiv preprint arXiv:2006.07397*, 2020. 1, 2

[4] Xuefeng Du, Zhaoning Wang, Mu Cai, and Sharon Li. Towards unknown-aware learning with virtual outlier synthesis. In *International Conference on Learning Representations*, 2022. 2

[5] Alexandros Haliassos, Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. Lips don't lie: A generalisable and robust approach to face forgery detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5039–5049, 2021. 4

[6] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3204–3213, 2020. 1

[7] Junting Pan, Ziyi Lin, Xiatian Zhu, Jing Shao, and Hongsheng Li. St-adapter: Parameter-efficient image-to-video transfer learning. *Advances in Neural Information Processing Systems*, 35:26462–26477, 2022. 1

[8] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. 1, 2

[9] Md Shohel Rana, Mohammad Nur Nobi, Beddhu Murali, and Andrew H Sung. Deepfake detection: A systematic literature review. *IEEE access*, 10:25494–25513, 2022. 1

[10] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Niessner. Faceforensics++: Learning to detect manipulated facial images. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1–11, 2019. 2

[11] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9 (86):2579–2605, 2008. 2

[12] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, and Houqiang Li. Altfreezing for more general video face forgery detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4129–4138, 2023. 4