Leveraging Failed Samples: A Few-Shot and Training-Free Framework for Generalized Deepfake Detection

Shibo Yao[1, 2], **Renshuai Tao**[1, 2*], **Xiaolong Zheng**[3], **Chao Liang**[4], **Chunjie Zhang**[1, 2]

[1]Institute of Information Science, Beijing Jiaotong University
[2]Visual Intelligence +X International Cooperation Joint Laboratory of MOE
[3]Institute of Automation, Chinese Academy of Sciences
[4]School of Computer Science, Wuhan University
{24125350, rstao}@bjtu.edu.cn

Abstract

Recent deepfake detection studies often treat unseen sample detection as a "zero-shot" task, training on images generated by known models but generalizing to unknown ones. A key real-world challenge arises when a model performs poorly on unknown samples, yet these samples remain available for analysis. This highlights that it should be approached as a "few-shot" task, where effectively utilizing a small number of samples can lead to significant improvement. Unlike typical few-shot tasks focused on semantic understanding, deepfake detection prioritizes image realism, which closely mirrors real-world distributions. In this work, we propose the Few-shot Training-free Network (FTNet) for real-world few-shot deepfake detection. Simple yet effective, FTNet differs from traditional methods that rely on large-scale known data for training. Instead, FTNet uses only one fake sample from an evaluation set, mimicking the scenario where new samples emerge in the real world and can be gathered for use, without any training or parameter updates. During evaluation, each test sample is compared to the known fake and real samples, and it is classified based on the category of the nearest sample. We conduct a comprehensive analysis of AI-generated images from 29 different generative models and achieve a new SoTA performance, with an average improvement of 8.7% compared to existing methods. This work introduces a fresh perspective on real-world deepfake detection: when the model struggles to generalize on a few-shot sample, leveraging the failed samples leads to better performance.

Code — https://github.com/zuiluorenjian/FTNet

Introduction

With the rapid development of deep learning-based techniques (Zhang et al. 2025), especially the rise of advanced generative models such as Generative Adversarial Networks (GANs) (Goodfellow et al. 2014; Rossler et al. 2019; Choi et al. 2018) and Diffusion models (Rombach et al. 2022a; Nichol et al. 2021b; Podell et al. 2023; Esser et al. 2024), AI-generated images have made significant strides in producing realistic and diverse content. These models, which have evolved over the years, are capable of generating highly convincing synthetic media, including faces and entire video sequences that are nearly indistinguishable from real.

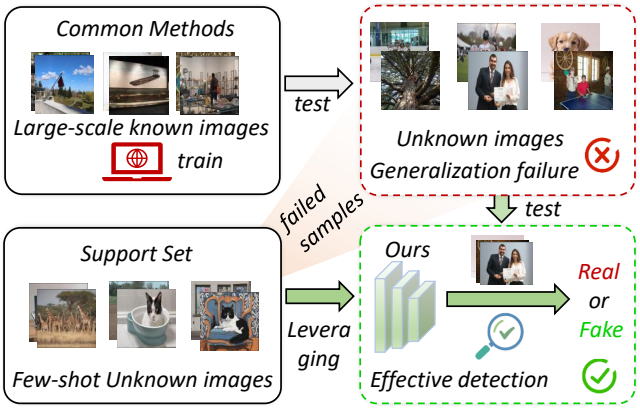

Figure 1: Comparison of traditional methods and our few-shot method in real-world scenarios.

However, AI-generated content, particularly deepfakes, has become a significant threat to information integrity, public trust, and digital security. Deepfake technology allows for the creation of hyper-realistic fake media, which can be easily shared and distributed, making it difficult for the public to distinguish between what is real and what is fabricated. The implications are far-reaching: deepfakes can be used for spreading false propaganda, manipulating public opinion, committing commercial fraud, or even defaming individuals by altering their appearance or words in a video.

Moreover, as the quality of generative models continues to improve, detecting deepfakes has become increasingly challenging. Traditional detection methods(Tan et al. 2024b; Yan et al. 2024b; Guillaro et al. 2025; Tao et al. 2025) that rely on identifying artifacts or inconsistencies in synthetic images and videos are gradually losing effectiveness as generative models become more sophisticated. For instance, deepfake videos created with state-of-the-art GANs or Diffusion models often exhibit a high level of realism, making it harder to distinguish them from real media. This creates a pressing need for deepfake detection systems that are not only capable of detecting known types of deepfakes but also gener-

*Corresponding author.

alize well to new, unseen variants. In response to this challenge, deepfake detection research has focused on developing methods that can adapt to novel deepfake samples. As shown in Figure 1. Many existing techniques treat the detection of unseen samples as a "zero-shot" task, in which models are trained on a set of known deepfakes and expected to generalize to unknown ones. However, this approach often falls short in real-world scenarios, where new deepfakes are continuously created, and the model's performance degrades when exposed to previously unseen content. This highlights the importance of developing more robust detection methods that can handle unseen deepfakes effectively, without relying on extensive retraining or large-scale datasets. As shown in Figure 2(a), the baseline detector overfits to the known generative model and has difficulty distinguishing unknown generative models. Instead, our method (Figure 2(b)) classifies various generative models well, which indicates its great potential in addressing the above challenges.

In this work, we introduce the Few-shot Training-free Network (FTNet), a novel approach for real-world deepfake detection. One promising direction in deepfake detection is the use of "few-shot" learning approaches, where the model is trained to recognize deepfakes from only a small number of examples. Few-shot learning can be particularly useful in real-world scenarios, where **new deepfake samples may emerge and can be quickly gathered for analysis**. Unlike traditional methods that require training on large amounts of labeled data, few-shot detection techniques can adapt to new deepfake types with minimal resources. However, deepfake detection using few-shot learning presents unique challenges, primarily because **deepfakes focus on mimicking the realism of images, which closely resemble real-world distributions**. This makes it difficult for conventional few-shot learning methods, which often focus on semantic understanding, to perform effectively in deepfake detection.

The proposed FTNet addresses these challenges by operating in a training-free manner. It uses only a minimal number of samples, specifically **one fake sample** from the evaluation set, mimicking real-world scenarios where new deepfakes emerge and can be quickly gathered for use. The features and labels of these samples are then injected into a dynamic knowledge base, which we call the key-value cache. This approach eliminates the need for parameter updates or retraining, significantly reducing computational overhead and resource requirements. During the evaluation phase, each test sample is compared against the known fake and real samples and is classified based on the category of the nearest sample. FTNet's simple yet effective method allows it to rapidly adapt to new deepfake samples, offering a scalable and computationally efficient solution for detecting.

Through extensive experiments on **AI-generated images from 29 different generative models**, we demonstrate that the proposed FTNet achieves new state-of-the-art performance in deepfake detection, outperforming existing methods by **8.7%** (with the fine-tuning version, FTNet-T, achieving 12.1%). This work highlights the potential of using a small-sample, training-free method to enhance deepfake detection, especially in scenarios where new deepfakes are constantly being generated. By focusing on effectively leveraging few-shot samples, FTNet provides a scalable and practical solution to the growing problem of deepfake detection. The main contributions are summarized as follows:

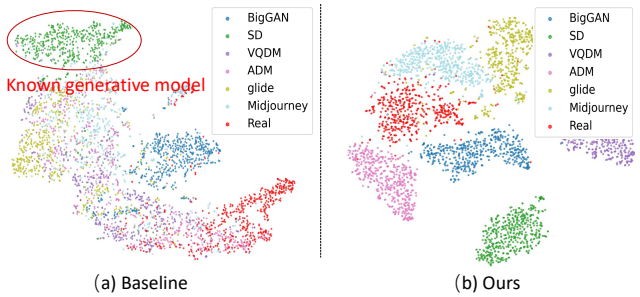

Figure 2: Comparison of traditional detector and our method on a 6-class generative model feature space.

- We propose the FTNet, a novel approach for deepfake detection that uses a minimal number of samples (only one fake and one real sample) without any need for retraining or parameter updates. This approach significantly reduces the need for large-scale data and training resources.
- FTNet is designed to handle real-world deepfake detection, where new, unseen deepfake samples frequently emerge. By utilizing a few-shot, training-free methodology, FTNet adapts to new deepfake types effectively, providing a practical solution in dynamic environments.
- Through extensive experiments on AI-generated images from 29 different generative models across three open-world datasets, demonstrating the effectiveness.

Related Work

Owing to space constraints in the main document, this section has been included in the supplementary materials.

Method

In this section, we will present its core components in order: the Cache Model Construction, our training-free FTNet, and its fine-tuned variant, FTNet-T.

Problem Definition

The core challenge faced by the deepfake detection field is that existing methods usually rely on large-scale data training of a single generative model (such as ProGAN(Karras et al. 2017) or a specific diffusion model(Ho et al. 2020)), which quickly becomes outdated in the context of the diversification and rapid evolution of AI generation technology. With the continuous emergence of new generative models (such as upgraded GAN variants, iterative versions of diffusion models, etc.), the forgery traces of their synthetic images are more hidden and show unique artifact characteristics, resulting in the serious lack of generalization ability of traditional methods due to the large domain difference between the source domain and the target domain. In addition, in real scenarios, we can often obtain a small number of samples of new generative models, while existing methods either

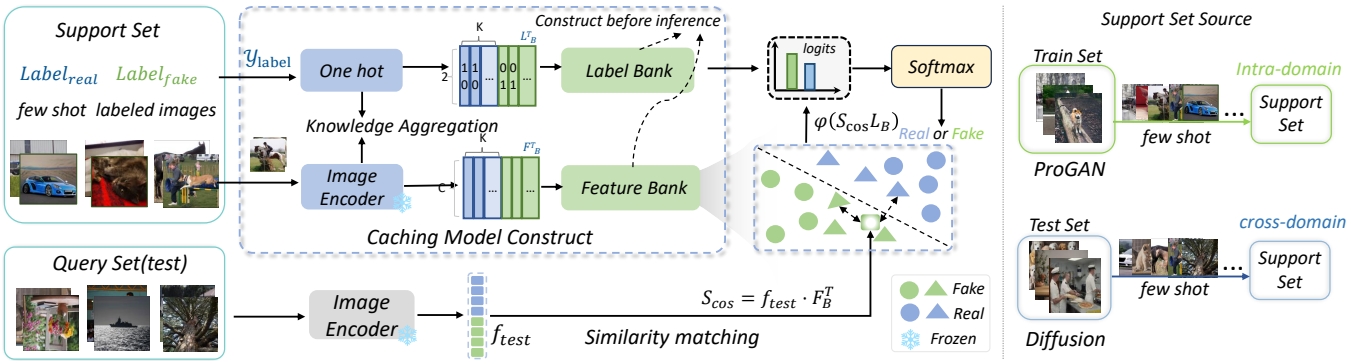

Figure 3: Overall framework of our proposed FTNet. The core of FTNet extracts features from the intermediate layers of the CLIP image encoder to build a cache module, leveraging few-shot labeled samples for efficient detection of unknown samples.

require a large amount of labeled data for training or cannot use limited samples to update detection capabilities due to the lack of dynamic adaptive mechanisms. By reconceptualizing deepfake detection as a few-shot task, that is, treating images from different generators as different categories, we can simplify this complex problem. This approach allows us to extract rich features from very few samples and achieve automatic adaptation to unseen biases without further large-scale fine-tuning or large datasets.

Framework Overview

To address the challenge of the lack of generalization ability of existing deepfake detections when facing unknown new models, we propose a novel few-shot detection framework. As shown in Figure 3. This framework uses the pre-trained CLIP model as the visual backbone, builds an efficient cache model, and uses a small number of target domain samples to identify the synthetic images of the newly generated model. It is suitable for open-world scenarios with scarce data. We first propose FTNet without parameter training, which identifies newly generated images by building a key-value cache. It is suitable for scenarios with extremely limited resources, real-time deployment, or extremely scarce samples. On this basis, we propose FTNet-T, which breaks through the performance ceiling by fine-tuning the cache model at low cost. It is suitable for scenarios that pursue high precision, have sufficient samples, or can be updated in non-real time. FTNet and FTNet-T form a complete paradigm of "adaptive knowledge injection" : FTNet lays the foundation for instant caching of new fake knowledge into the CLIP feature space, and FTNet-T optimizes knowledge through lightweight fine-tuning to achieve a seamless transition from basic adaptation to precise detection, ensuring the efficiency of the framework in a variety of practical scenarios.

Cache Model Construction

Past few-shot deepfake detection methods(Wu et al. 2025) treated different generative models as different categories, and unified real images into one category. However, this setting results in them only being able to test a single dataset during the test, which is not in line with real scenarios. Real-world detection scenarios are often characterized by a mixture of images from multiple, diverse, and often unknown sources. In our task, although we treat the generative model as an independent category, we also ensure that the total number of true and false images in the cached model is balanced, that is, we extract the same real and fake images from each dataset. It also simulates a simulated real mixed environment, that is, the number of real images is greater than that of a single generative model. Specifically, the few images we extract from these datasets are regarded as the support set $\mathcal{D}_{support} = \{(x_i, y_i)\}_{i=1}^{N}$ for building our cached model. A set of images to be tested is regarded as a query set $\mathcal{D}_{query} = \{(x_q)\}_{q=1}^{N_q}$. This task aims to predict the category of each image in the query set based on the information provided by the support set.

We aim to enhance the model's capability to detect synthetic images by integrating a small amount of new knowledge, eliminating the need for traditional large-scale training. Leveraging a pre-trained CLIP model and a limited number of dataset samples from the target domain, we strategically extract image features from the intermediate layers of the CLIP to effectively capture forgery traces in synthetic images. Our proposed cache model comprises two core components: a Feature Bank and a Label Bank. Specifically, we utilize the CLIP to extract D-dimensional features normalized via L_2 normalization $f_k \in \mathbb{R}^D$. The true label $y_k \in \{\text{real}, \text{fake}\}$ corresponding to each image x_k is converted into a 2-dimensional one-hot encoding vector $l_k \in \mathbb{R}^2$. The f_k and l_k are as follows:

$$f_k = \frac{E_{vis}(x_k)}{\|E_{vis}(x_k)\|} \tag{1}$$

$$l_k = \text{OneHot}(y_k) \tag{2}$$

These extracted intermediate layer features f_k are treated as keys, aggregated into Feature Bank, denoted as F_B. The one-hot ground-truth vectors l_k are used as their values, aggregated into Label Bank. In the form of key-value pairs, the entire cache model can be expressed as $(\mathcal{K}, \mathcal{V}) = (\{f_k\}_{k=1}^{N_S}, \{l_k\}_{k=1}^{N_S})$ This cached model is built before the inference phase begins; in this way, the model remembers all new knowledge extracted from a few samples.

Few-shot Training-free Network (FTNet)

For an unseen test image, we use the CLIP intermediate layer to extract global features f_{test} , which are in the same embedding space as the features in the feature bank. We only need two simple matrix-vector multiplications to achieve classification. First, we can calculate the cosine similarity between the test image features and F_B by

$$S_{\cos} = \frac{f_{\text{test}} F_B^T}{\|f_{\text{test}}\| \cdot \|F_B\|} \in \mathbb{R}^{1\times N} \tag{3}$$

This represents the semantic relevance of the test image to the images in the cache. Then, use $S_{\cos}$ as the weight to integrate the one-hot encoded labels in the label bank L_B to obtain the classification logits $\in \mathbb{R}^{1\times 2}$. The process can be formalized as follows:

$$\text{logits} = \varphi(S_{\cos} L_B) \tag{4}$$

where $\varphi(x) = \exp(-\alpha(1-x))$ is an activation function(Zhang et al. 2021). α represents a tuning hyperparameter. The exponential function converts the similarity into a non-negative value and uses α to modulate its sharpness. In $S_{\cos}$ similar feature memories with higher scores contribute more to the final classification logits, and vice versa. Through this similarity-based label integration, FTNet can adaptively distinguish synthesized images.

FTNet with Fine-tuning (FTNet-T)

FTNet can quickly identify new AI synthetic images by caching a small amount of target domain sample knowledge. However, as the number of cached samples increases, the generalization ability of the model tends to be flat. In order to further improve the generalization ability, we propose the FTNet-T, which fine-tunes a small number of samples in the cache, that is, fine-tunes the keys in the Feature Bank. In view of the extreme scarcity of new generative model samples in the real world, FTNet-T achieves advanced performance on advanced forged image datasets with a fine-tuning cost as low as 20 epochs.

Specifically, we unfreeze the keys in the Feature Bank, keep the values in the Label Bank frozen, and add a linear layer $L_A(\cdot)$ to perform learning. The similarity is calculated by the output of the linear layer. The formula is as follows:

$$S_{\cos} = L_A(f_{test}) = f_{test} W_A \tag{5}$$

Its internal learnable weight matrix is $\mathbf{W}_A \in \mathbb{R}^{D\times N}$. $\mathbf{W}_A$ is initialized as F_{B}^T. However, it will be updated according to the task objectives. During fine-tuning, cross-entropy loss is used as the main objective function to measure the difference between the model prediction and the true label.

Experiments

In this section, we evaluate the performance of our FTNet and FTNet-T methods for few-shot deepfake detection through a series of experiments.

Settings

Datasets: To ensure the consistency of the benchmark, we used 3 benchmark datasets for implementation. They are **GenImage**, **UniversalFakeDetect**, and **OpenSDI** dataset. The GenImage million-level dataset contains 8 generators (7 Diffusion models and one GAN).The generators include Midjourney(Midjourney 2022), Stable Diffusion V1.4(Rombach et al. 2022b), StableDiffusionV1.5(Rombach et al. 2022b), Wukong(Wukong 2022), ADM(Dhariwal et al. 2021), GLIDE(Nichol et al. 2021a), VQDM(Gu et al. 2022) and BigGAN(Brock et al. 2018). The UniversalFakeDetect(Ojha et al. 2023) includes 20 subsets of generated images. The test set comprises 19 subsets from various generative models including ProGAN(Karras et al. 2017), StyleGAN(Karras et al. 2019), BigGAN(Brock et al. 2018), CycleGAN(Zhu et al. 2017), StarGAN(Choi et al. 2018), GauGAN(Park et al. 2019)and Deepfake(Rossler et al. 2019), CRN(Chen et al. 2017), IMLE(Li et al. 2019), SAN(Dai et al. 2019), SITD(Chen et al. 2018), Guided(Dhariwal et al. 2021), LDM(Rombach et al. 2022b), Glide(Nichol et al. 2021a), DALLE(Ramesh et al. 2021). The OpenSDI(Wang et al. 2025) open world complex dataset contains 5 generative models. The generators include SD1.5(Rombach et al. 2022b), SD2.1(Rombach et al. 2022b), SDXL(Podell et al. 2023), SD3(Esser et al. 2024) and Flux.1(Labs 2024). Since we regard different generative models as different categories and real images as a separate category, following the settings of Wu et al. (Wu et al. 2025), the three diffusion models SDv1.4, SDv1.5 and WuKong in the GenImage have the same model structure and are difficult to distinguish, so the three models are unified into the SD model.

Implementation Details: Our experiments are conducted under 4-shot and 8-shot few-shot learning configurations. We define k-shot as follows: to build the cache model, we randomly sample k real and k fake images from each generative model's dataset. This sampling strategy ensures that the total number of real images in the cache (summed across all sources) is greater than the number of fake images from any single generator, which closely simulates real-world scenarios. For our model architecture, we utilize a pretrained CLIP model with a ViT-L/14 backbone and set the input image resolution to 224x224 pixels. Regarding hyperparameters, the temperature parameter α in the FTNet module's activation function is set to 15. For the fine-tunable FTNet, we employ an AdamW (Kingma et al. 2014) optimizer with a learning rate of 0.001 and the training epochs are 20. The primary evaluation metric is Accuracy (Acc). Additionally, to ensure a comprehensive comparison on the OpenSDI dataset (Wang et al. 2025), we also report the F1-score.

Comparisons with State-of-the-art Methods

Evaluation on Genimage. To fully evaluate our framework on the GenImage, we adopt two common schemes, as shown in Table 1 and Table 2. These schemes are designed to evaluate different aspects of the model's generalization ability, from standard single-source migration to strict cross-generator robustness. The results of accuracy (Acc) are pre-

Method	Intra-domain							Cross-domain						
	Mid	SD	BigGAN	ADM	VQDM	GLIDE	**mAcc**	Mid	SD	BigGAN	ADM	VQDM	GLIDE	**mAcc**
GramNet (2020)	75.8	81.8	93.1	58.6	76.2	88.9	79.0	54.2	99.0	51.7	50.3	50.8	54.6	69.9
CNNSpot (2020)	88.0	92.3	98.9	81.2	93.1	95.8	91.5	52.8	90.2	46.8	50.1	53.4	39.8	64.2
SBI (2022)	87.9	87.1	84.5	84.4	86.7	97.1	87.9	66.8	99.2	50.0	50.6	49.8	55.8	62.0
F3Net (2020)	90.5	89.6	95.0	84.1	93.3	98.1	91.8	50.1	99.9	49.9	49.9	49.9	50.0	68.7
UnivFD (2023)	85.5	85.1	89.1	96.9	86.7	88.3	88.6	93.9	95.6	90.5	71.9	81.6	85.4	88.8
NPR (2024b)	89.6	91.2	98.7	92.7	91.5	95.2	**93.1**	81.0	97.6	84.2	76.9	84.1	89.8	88.6
FreqNet (2024a)	85.8	77.6	98.9	90.9	84.8	89.2	87.8	89.6	98.2	81.4	66.8	75.8	86.5	86.8
AIDE (2024a)	90.1	90.7	95.5	92.9	87.4	93.8	91.7	79.4	99.4	66.9	78.5	80.3	91.8	86.9
FTNet (4-shot)	93.8	91.8	94.5	85.8	78.7	94.6	89.9	93.4	86.9	97.1	90.0	80.2	96.6	90.7
FTNet-T (4-shot)	94.9	92.9	95.6	90.7	84.1	95.5	92.3	92.1	93.1	96.5	94.7	92.5	96.1	**94.2**

Table 1: Comparison under a Single-Source Training Scheme. In cross-domain testing, results are taken from original papers (Tan et al. 2025; Yan et al. 2024a). Baseline methods are trained on SDv1.4. Midjourney, denoted as Mid. **Bold** and underline indicate the best and second-best performances.

Method	Mid	SD	BigGAN	ADM	VQDM	GLIDE	**mAcc**
GramNet (2020)	58.1	72.8	61.2	58.7	57.8	65.3	62.3
CNNSpot (2020)	58.2	70.3	56.6	57.0	56.7	57.1	59.3
F3Net (2020)	55.1	73.1	56.5	66.5	62.1	57.8	61.9
UniFD (2023)	70.8	74.6	86.1	70.0	71.9	73.2	74.4
DIRE (2023)	65.0	73.7	56.7	61.9	63.4	69.1	65.0
LARE2 (2024)	66.4	87.3	74.0	66.7	84.4	81.3	76.7
NPR (2024b)	74.6	76.3	83.2	70.6	64.9	89.2	76.5
FSD (zero) (2025)	75.1	88.0	62.1	74.1	69.1	93.9	77.1
FSD (10) (2025)	80.9	88.8	82.2	79.2	76.2	97.1	84.1
LDC (10) (2025)	59.1	70.8	80.8	72.6	80.7	78.5	73.8
FTNet (**ours**)	93.4	86.9	97.1	90.0	80.2	96.6	90.7
FTNet-T (**ours**)	92.1	93.1	96.5	94.7	92.5	96.1	**94.2**

Table 2: Comparison with Cross-Generator Validation Scheme. We implemented UniFD and NPR while other baselines are cited from (Wu et al. 2025).

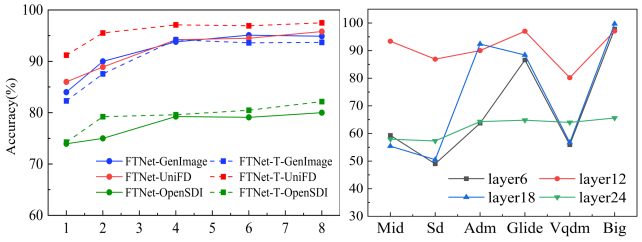

Figure 4: Analysis of the method's performance. (Left) Influences of the number of shots. (Right) Performance of the FTNet method on different CLIP layers.

sented. First, Table 1 evaluates the single-source generalization ability, including two benchmarks: in-domain and cross-domain testing. In the in-domain experiment, the training and test generator types are consistent. We reduced the training set of the baseline method. Each method's training set is 2K images (real and fake ratio 1:1), and the official experimental setting is used for training. In our intra-domain experiment, we randomly sample from the original training set. Our FTNet-T achieves excellent results in the 4-shot intra-domain test. Its average accuracy reaches 92.3%, ranking second among all methods and significantly improving the performance by 3.7% compared to the baseline method UniFD. In the cross-domain test, all baseline methods are trained on the original SDv1.4 dataset, while our method bypasses large-scale training and only needs a few samples from the target domain to achieve SOTA performance. In the 4-shot setting, our method FTNet improves 1.9% over UniFD, and 5.4% over the baseline when we further fine-tune. These results highlight that our method can be directly applied to detection tasks without any training, and its performance is superior to traditional methods that rely on large-scale data training. Table 2 adopts the cross-generator verification scheme. This scheme trains 6 specialized models on 6 independent generators for the baseline method and calculates their average performance on unseen data. This scheme clearly reveals the limitations of traditional zero-shot methods when facing unknown generative models; that is, when the test data differs too much from the training source, its performance will seriously decline. As shown in the results in the table, our method effectively alleviates this problem: untrained FTNet improves by 6.6% compared to FSD (10-shot), while FTNet-T improves by 10.1%.

Evaluation on UniversalFakeDetect. The accuracy (Acc) results are shown in Table3. UniFD is similar to our method in that it retains the original pre-trained knowledge of CLIP and only achieves classification by fine-tuning the fully connected layer. Compared with UniFD, our method without training improves mAcc by 12.65%. When we adopt light fine-tuning, mAcc improves by 15.76%. In addition, compared to the latest state-of-the-art method, C2P-clip, our method improves accuracy by 4.14%, which proves the superiority of our method without the need for text encoders and large sample training.

Evaluation on OpenSDI. The accuracy (Acc) results are listed in Table 4. The OpenSDI is designed to simulate the image forgery detection challenge in the open world. The

Method	GAN						Deep fakes	Low level		Perc.loss		Guided	LDM			Glide			Dalle	**mAcc**
	Pro GAN	Cycle GAN	Big GAN	Style GAN	Gau GAN	Star GAN		SITD	SAN	CRN	IMLE		200 steps	200 w/cfg	100 steps	100 27	50 27	100 10		
CNN-Spot	99.99	85.20	70.20	85.70	78.95	91.70	53.47	66.67	48.69	86.31	86.26	60.07	54.03	54.96	54.14	60.78	63.80	65.66	55.58	69.58
Freq-spec	49.90	99.90	53.50	49.90	50.30	99.70	50.10	50.00	48.00	50.60	50.10	50.90	50.40	50.40	50.30	51.70	51.40	50.40	50.00	55.45
F3Net	99.38	76.38	65.33	92.56	58.10	100.00	63.48	54.17	47.26	51.47	51.47	69.20	68.15	75.35	68.80	81.65	83.25	83.05	66.30	71.33
UniFD	100.00	98.50	94.50	82.00	99.50	97.00	66.60	63.05	57.50	59.50	72.00	70.03	94.19	73.76	94.36	79.07	79.85	78.14	78.68	81.38
LGrad	99.84	85.39	82.88	94.83	72.45	99.62	58.00	62.50	50.00	50.74	50.78	77.50	94.20	95.85	94.80	87.40	90.70	89.55	88.35	80.28
FreqNet	97.90	95.84	90.45	97.55	90.24	93.41	97.40	88.82	59.04	71.92	67.35	86.70	84.55	99.58	65.56	85.69	97.40	88.15	59.06	85.09
NPR	99.84	95.00	87.55	96.23	86.57	99.75	76.90	66.94	98.63	50.00	50.00	84.55	97.65	98.00	98.20	96.25	97.15	97.35	87.15	87.56
FatFormer	99.89	99.32	99.50	97.15	99.41	99.75	93.23	81.11	68.04	90.45	90.45	76.00	98.60	94.90	98.65	94.35	94.65	94.20	98.75	90.86
C2P-CLIP	99.71	90.69	95.28	99.38	95.26	96.60	89.86	98.33	64.61	90.69	90.69	77.80	99.05	98.05	98.05	94.65	94.20	94.40	98.80	93.00
FTNet(4)	98.37	96.17	76.18	99.55	75.63	100.00	83.38	83.24	75.58	99.97	99.97	98.80	100.00	100.00	100.00	99.85	99.90	100.00	100.00	<u>94.03</u>
FTNet-T	99.69	99.20	97.47	99.77	97.85	100.00	85.45	98.58	76.05	99.95	99.95	93.17	100.00	99.95	100.00	99.35	99.55	99.60	99.95	**97.14**

Table 3: Comparison on the UniversalFakeDetect. All results are copied from C2p-CLIP(2025), including the reported mAcc of Freq-spec(2019) and FatFormer(2024). **Bold** and <u>underline</u> indicate the best and second-best performances, respectively.

Method	SD1.5		SD2.1		SDXL		SD3		Flux.1		**AVG**	
	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	mAcc
GramNet (Liu et al. 2020)	80.51	80.35	74.01	76.66	65.28	70.76	64.35	70.29	52.00	63.37	67.23	72.29
CNNSpot (Wang et al. 2020)	84.60	85.04	71.56	75.94	59.70	68.72	56.27	67.08	35.72	57.57	61.57	70.87
MVSS-Net (Chen et al. 2021)	93.47	93.65	79.27	82.33	59.85	70.42	62.80	72.13	27.59	56.78	64.60	75.06
ObjectFormer (Wang et al. 2022)	71.72	75.22	66.79	72.55	49.19	62.92	48.32	62.54	37.92	58.05	54.79	66.26
UniFD (Ojha et al. 2023)	77.45	77.60	80.62	81.92	70.74	74.83	71.09	75.17	61.10	69.06	72.20	75.72
FreqNet (Tan et al. 2024a)	75.88	77.70	60.97	68.37	53.15	64.02	53.50	64.37	38.47	57.08	56.39	66.31
IML-ViT (Ma et al. 2023)	94.47	75.73	69.70	61.19	40.98	49.95	44.69	51.25	18.20	43.62	53.61	56.35
NPR (Tan et al. 2024b)	79.41	79.28	81.67	81.84	72.12	74.28	73.43	75.47	67.62	71.36	74.85	76.45
RINE (Koutlis et al. 2024)	91.08	90.98	87.47	88.12	73.43	78.76	72.05	76.78	55.86	67.02	75.98	80.33
MaskCLIP (Wang et al. 2025)	92.64	92.72	88.71	89.45	78.02	81.22	73.07	78.01	56.49	68.50	77.79	<u>81.98</u>
FTNet (**8-shot**)	73.24	74.62	80.17	82.21	82.14	83.63	76.58	79.64	82.57	79.58	<u>77.83</u>	79.94
FTNet-T (**8-shot**)	78.04	77.70	84.13	84.79	85.82	86.17	84.06	84.69	81.39	82.54	**82.68**	**83.16**

Table 4: Comparison on the OpenSDI. The baseline methods results are directly cited from (Wang et al. 2025). The baseline methods compared are all trained on SD1.5 (100k real + 100k fake).

construction of this dataset fully considers the three core open-world settings of "user diversity", "model innovation", and "operation range", making it a more challenging and realistic benchmark. Compared with the baseline UniFD, FTNet improves the accuracy by 4.22%, and the accuracy is improved by 7.44% after further fine-tuning. In addition, compared to the MaskClip, FTNet-T improved accuracy by 1.18%, but MaskClip combines the visual and text encoders of CLIP and a masked autoencoder (MAE)(He et al. 2022) encoder, which relies on high-performance computing facilities and is very time-consuming.

Ablation Study

The number of shots. To determine the minimum number of samples required for this method to achieve the best balance between performance and computational overhead, we conduct an ablation experiment. We examine the corresponding performance of the model when taking different sampling numbers. We test it on three major datasets, named GenImage, UniFD (UniversalFakeDetect), and OpenSDI datasets. The experimental results are shown in Figure 4 (left). The performance improves with the increase in the number of samples, but this growth is not linear. When the number of samples reaches a certain scale, the performance gain becomes very limited. Our method has the best overall performance under 4-shot sampling.

Method	Sampling	GenImage	UniFD	OpenSDI	**mAcc**
UniFD(2023)	8-shot	70.0	85.9	52.1	69.3
NPR(2024b)	8-shot	78.8	87.4	55.1	73.8
FTNet(**ours**)	4-shot	90.7	89.3	79.3	86.4
FTNet-T(**ours**)	4-shot	**94.2**	**97.1**	**79.6**	**90.3**

Table 5: Comparison with fine-tuning-based methods.

Few-shot Evaluation with Fine-tuning Baselines. To evaluate the performance of our approach, we conducted a comparative experiment. We fine-tuned the baseline models UniFD and NPR using 8-shot training and compared the results with ours, which utilized 4-shot samples, as shown in Table 5. The experimental results show that, even with

Method	Training samples	Target domain					mAcc
		ADM	BigGAN	GLIDE	VQDM	Midj	
UniFD	320k	71.9	**90.5**	85.4	81.6	**93.9**	84.6
NPR	320k	76.9	84.2	89.8	**84.1**	81.0	83.2
FTNet	500	73.9	85.5	85.2	70.9	84.3	80.8
FTNet-T	500	**78.3**	88.8	**94.4**	75.8	92.5	**86.0**

Table 6: Comparison with zero-shot generalization methods.

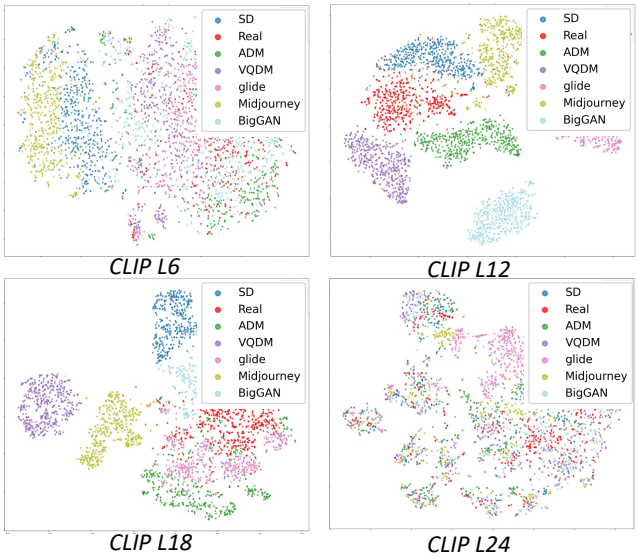

Figure 5: Visualization of FTNet Using CLIP ViT-L/14 Features (L6, L12, L18, L24) on the GenImage Dataset.

4-shot samples, FTNet achieves 86.4% mAcc, significantly outperforming both 8-shot baselines. The fine-tuned FTNet-T further improves performance to 90.3% mAcc. This result significantly outperforms the 8-shot fine-tuned UniFD and NPR by 21.0% and 16.5%, respectively. This comparison shows the superior performance of our approach.

Zero-shot Generalization Performance Comparison. To evaluate the zero-shot generalization capability of our method, we compare it with two baseline methods, UniFD and NPR, and the results are shown in Table 6. Our model shows a significant advantage in data efficiency. Specifically, both UniFD and NPR are trained on the SDv1.4 dataset, while our FTNet-T is fine-tuned on only 0.15% of the data, achieving an improvement in accuracy of about 1.4% and 2.8% over UniFD and NPR, respectively. This shows the strong potential of our method to learn forgery patterns.

Generalization Ability of CLIP Feature Layers. We explore how different feature layers within the CLIP affect the generalization ability of our method. We utilize FTNet and analyze features from four different layers: the 6th, 12th, 18th, and 24th layers (L6, L12, L18, L24) in a 4-shot setting. The results are shown in Figure 4 (right). The detection performance is not simply linear with the layer depth. Among them, the middle-level features from L12 achieve the highest average accuracy , outperforming all other layers. CLIP's L12 strikes the best balance between preserving these critical low-level artifact information and learning high-level representations that distinguish between different generator models. To visually verify these findings, the t-SNE of the features is shown in Figure 5. The feature clusters at L12 are the clearest and most compact, while the features at the shallower L6 are mixed. The excessively deep L24 is overly abstract, making it nearly impossible to distinguish between genuine and forged features.

Figure 6: The top and bottom rows display two query cases and their nearest neighbors in the feature space.

Feature Space Visualization and Nearest Neighbor Analysis. To demonstrate the decision-making process of our method, we conduct a qualitative analysis. As shown in the Figure 6. We randomly selected some images to construct a support set. The experiments revealed an interesting phenomenon: when a forged query image (e.g., an astronaut) is input, its nearest neighbor in feature space is a completely unrelated forged image (e.g., a bus or a giraffe) rather than a real image with similar content. This finding strongly demonstrates that the CLIP intermediate-level features relied upon by our method successfully capture a universal forgery signature across semantic content.

Conclusion

In this work, we address a critical challenge in deepfake detection by rethinking the task as a few-shot problem rather than a traditional "zero-shot" task. While many studies treat unseen sample detection as a generalization issue, we highlight that real-world scenarios demand more effective utilization of limited samples, improving performance on previously unseen deepfakes. Our proposed few-shot method (FTNet) provides **a novel solution by leveraging only one fake sample from the evaluation set**, mimicking real-world conditions where new samples are gathered without requiring retraining or parameter updates. FTNet outperforms traditional methods that depend on large-scale training datasets and achieves state-of-the-art performance, with an 8.7% improvement on average compared to existing methods. By incorporating this new perspective on few-shot detection, we show that effectively using failed samples in real-world deepfake detection can significantly improve performance. This work provides a step toward more practical solutions for detection in dynamic environments.

Acknowledgments

This work is supported by the Natural Science Foundation of China (No.62476021, No.U24B20179, No.72434005, No.72225011 and No.62372339), the Fundamental Research Funds for the Central Universities (2025JBZX062), Key Science and Technology Research Project of Xinjiang Production and Construction Corps (2025AB029).

References

Brock, A.; Donahue, J.; and Simonyan, K. 2018. Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*.

Chen, C.; Chen, Q.; Xu, J.; and Koltun, V. 2018. Learning to see in the dark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3291–3300.

Chen, Q.; and Koltun, V. 2017. Photographic image synthesis with cascaded refinement networks. In *Proceedings of the IEEE international conference on computer vision*, 1511–1520.

Chen, X.; Dong, C.; Ji, J.; Cao, J.; and Li, X. 2021. Image manipulation detection by multi-view multi-scale supervision. In *Proceedings of the IEEE/CVF international conference on computer vision*, 14185–14193.

Choi, Y.; Choi, M.; Kim, M.; Ha, J.-W.; Kim, S.; and Choo, J. 2018. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8789–8797.

Dai, T.; Cai, J.; Zhang, Y.; Xia, S.-T.; and Zhang, L. 2019. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11065–11074.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Esser, P.; Kulal, S.; Blattmann, A.; Entezari, R.; Müller, J.; Saini, H.; Levi, Y.; Lorenz, D.; Sauer, A.; Boesel, F.; et al. 2024. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Gu, S.; Chen, D.; Bao, J.; Wen, F.; Zhang, B.; Chen, D.; Yuan, L.; and Guo, B. 2022. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10696–10706.

Guillaro, F.; Zingarini, G.; Usman, B.; Sud, A.; Cozzolino, D.; and Verdoliva, L. 2025. A bias-free training paradigm for more general ai-generated image detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 18685–18694.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16000–16009.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Karras, T.; Aila, T.; Laine, S.; and Lehtinen, J. 2017. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.

Karras, T.; Laine, S.; and Aila, T. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4401–4410.

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

Koutlis, C.; and Papadopoulos, S. 2024. Leveraging representations from intermediate encoder-blocks for synthetic image detection. In *European Conference on Computer Vision*, 394–411. Springer.

Labs, B. F. 2024. FLUX. https://github.com/black-forest-labs/flux.

Li, K.; Zhang, T.; and Malik, J. 2019. Diverse image synthesis from semantic layouts via conditional imle. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4220–4229.

Li, S.; Liu, F.; Hao, Z.; Wang, X.; Li, L.; Liu, X.; Chen, P.; and Ma, W. 2025. Logits DeConfusion with CLIP for Few-Shot Learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, 25411–25421.

Liu, H.; Tan, Z.; Tan, C.; Wei, Y.; Wang, J.; and Zhao, Y. 2024. Forgery-aware adaptive transformer for generalizable synthetic image detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10770–10780.

Liu, Z.; Qi, X.; and Torr, P. H. 2020. Global texture enhancement for fake face detection in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8060–8069.

Luo, Y.; Du, J.; Yan, K.; and Ding, S. 2024. LaREˆ 2: Latent reconstruction error based method for diffusion-generated image detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17006–17015.

Ma, X.; Du, B.; Jiang, Z.; Hammadi, A. Y. A.; and Zhou, J. 2023. IML-ViT: Benchmarking image manipulation localization by vision transformer. *arXiv preprint arXiv:2307.14863*.

Midjourney. 2022. Available at https://www.midjourney.com/home/.

Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2021a. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*.

Nichol, A. Q.; and Dhariwal, P. 2021b. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, 8162–8171. PMLR.

Ojha, U.; Li, Y.; and Lee, Y. J. 2023. Towards universal fake image detectors that generalize across generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24480–24489.

Park, T.; Liu, M.-Y.; Wang, T.-C.; and Zhu, J.-Y. 2019. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2337–2346.

Podell, D.; English, Z.; Lacey, K.; Blattmann, A.; Dockhorn, T.; Müller, J.; Penna, J.; and Rombach, R. 2023. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*.

Qian, Y.; Yin, G.; Sheng, L.; Chen, Z.; and Shao, J. 2020. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *European conference on computer vision*, 86–103. Springer.

Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; and Sutskever, I. 2021. Zero-shot text-to-image generation. In *International conference on machine learning*, 8821–8831. Pmlr.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022a. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022b. High-Resolution Image Synthesis With Latent Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10684–10695.

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.

Shiohara, K.; and Yamasaki, T. 2022. Detecting Deepfakes with Self-Blended Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18720–18729.

Tan, C.; Tao, R.; Liu, H.; Gu, G.; Wu, B.; Zhao, Y.; and Wei, Y. 2025. C2p-clip: Injecting category common prompt in clip to enhance generalization in deepfake detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 7184–7192.

Tan, C.; Zhao, Y.; Wei, S.; Gu, G.; Liu, P.; and Wei, Y. 2024a. Frequency-aware deepfake detection: Improving generalizability through frequency space domain learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 5052–5060.

Tan, C.; Zhao, Y.; Wei, S.; Gu, G.; Liu, P.; and Wei, Y. 2024b. Rethinking the up-sampling operations in cnn-based generative network for generalizable deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 28130–28139.

Tao, R.; Tan, C.; Liu, H.; Wang, J.; Qin, H.; Chang, Y.; Wang, W.; Ni, R.; and Zhao, Y. 2025. SAGNet: Decoupling Semantic-Agnostic Artifacts from Limited Training Data for Robust Generalization in Deepfake Detection. *IEEE Transactions on Information Forensics and Security*.

Wang, J.; Wu, Z.; Chen, J.; Han, X.; Shrivastava, A.; Lim, S.-N.; and Jiang, Y.-G. 2022. Objectformer for image manipulation detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2364–2373.

Wang, S.-Y.; Wang, O.; Zhang, R.; Owens, A.; and Efros, A. A. 2020. CNN-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8695–8704.

Wang, Y.; Huang, Z.; and Hong, X. 2025. OpenSDI: Spotting Diffusion-Generated Images in the Open World. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 4291–4301.

Wang, Z.; Bao, J.; Zhou, W.; Wang, W.; Hu, H.; Chen, H.; and Li, H. 2023. Dire for diffusion-generated image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 22445–22455.

Wu, S.; Liu, J.; Li, J.; and Wang, Y. 2025. Few-Shot Learner Generalizes Across AI-Generated Image Detection. *arXiv preprint arXiv:2501.08763*.

Wukong. 2022. Available at https://xihe.mindspore.cn/modelzoo/wukong.

Yan, S.; Li, O.; Cai, J.; Hao, Y.; Jiang, X.; Hu, Y.; and Xie, W. 2024a. A sanity check for ai-generated image detection. *arXiv preprint arXiv:2406.19435*.

Yan, Z.; Wang, J.; Jin, P.; Zhang, K.-Y.; Liu, C.; Chen, S.; Yao, T.; Ding, S.; Wu, B.; and Yuan, L. 2024b. Orthogonal Subspace Decomposition for Generalizable AI-Generated Image Detection. *arXiv preprint arXiv:2411.15633*.

Zhang, C.; Liu, C.; Duan, S.; Zheng, X.; Yu, T.; and Zhang, J. 2025. Embodied cognitive intelligence guided Moon sample collection. *The Innovation*.

Zhang, R.; Fang, R.; Zhang, W.; Gao, P.; Li, K.; Dai, J.; Qiao, Y.; and Li, H. 2021. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*.

Zhang, X.; Karaman, S.; and Chang, S.-F. 2019. Detecting and simulating artifacts in gan fake images. In *2019 IEEE international workshop on information forensics and security (WIFS)*, 1–6. IEEE.

Zhu, J.-Y.; Park, T.; Isola, P.; and Efros, A. A. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, 2223–2232.