Adapt to Scarcity: Few-Shot Deepfake Detection via Low-Rank Adaptation

Silvia Cappelletti[1,*], Lorenzo Baraldi[2,*], Federico Cocchi[1,2,*], Marcella Cornia[1], Lorenzo Baraldi[1], and Rita Cucchiara[1]

[1] University of Modena and Reggio Emilia, Italy
name.surname@unimore.it
[2] University of Pisa, Italy
name.surname@phd.unipi.it

Abstract. The boundary between AI-generated images and real photographs is becoming increasingly narrow, thanks to the realism provided by contemporary generative models. Such technological progress necessitates the evolution of existing deepfake detection algorithms to counter new threats and protect the integrity of perceived reality. Although the prevailing approach among deepfake detection methodologies relies on large collections of generated and real data, the efficacy of these methods in adapting to scenarios characterized by data scarcity remains uncertain. This obstacle arises due to the introduction of novel generation algorithms and proprietary generative models that impose restrictions on access to large-scale datasets, thereby constraining the availability of generated images. In this paper, we first analyze how the performance of current deepfake methodologies, based on the CLIP embedding space, adapt in a few-shot situation over four state-of-the-art generators. Being the CLIP embedding space not specifically tailored for the task, a fine-tuning stage is desirable, although the amount of data needed is often unavailable in a data scarcity scenario. To address this issue and limit possible overfitting, we introduce a novel approach through the Low-Rank Adaptation (LoRA) of the CLIP architecture, tailored for few-shot deepfake detection scenarios. Remarkably, the LoRA-modified CLIP, even when fine-tuned with merely 50 pairs of real and fake images, surpasses the performance of all evaluated deepfake detection models across the tested generators. Additionally, when LoRA CLIP is benchmarked against other models trained on 1,000 samples and evaluated on generative models not seen during training it exhibits superior generalization capabilities.

Keywords: Deepfake Detection · Few-Shot Learning · LoRA

1 Introduction

With the recent emergence of diffusion models [26,49] and the related enhancement in image quality, the text-to-image generative framework has facilitated

[*] Equal contribution.

the production of very realistic images from textual descriptions [5,41,42]. While this technology has enabled a wider distribution of artistic ability, it has also raised concerns about the spread of misinformation and social manipulation. To counter these side effects, deepfake detection emerges as a critical task aimed at identifying images that have been generated or altered by generative models.

Initial research in deepfake detection has mainly concentrated on identifying counterfeit faces [32,44]. Sequentially, different studies have expanded their scope to include the detection of natural images, considering a broader interest in ensuring the authenticity of a wide range of visual content. In this context, the CLIP (Contrastive Language-Image Pre-training) backbone [40] has been established as one of the most effective feature extraction methodologies for deepfake detection. Notably, when coupled with classification algorithms such as the k-Nearest Neighbor (k-NN), Support Vector Machines (SVMs), or linear classifiers, CLIP has demonstrated remarkable capabilities in discerning between generated and authentic content [1,14,36]. However, these solutions rely on large datasets comprising both real and generated images that may not be readily accessible with future generative models or commercial platforms [2,45]. Consequently, the effectiveness of CLIP-based detectors in scenarios characterized by limited data availability is still unclear and only partially approached in existing literature [14]. Further, despite the pre-trained CLIP embedding space demonstrating an ability to identify discriminative features relevant to deepfake detection, it is important to acknowledge that CLIP is optimized for a different task. For this reason, the adaptation of CLIP embedding space in the task of deepfake detection may result in improved classification results.

Low-Rank Adaptation (LoRA) [27], which originates for parameter efficient fine-tuning of large language models [16,28], has demonstrated its effectiveness in various tasks [7,8,39,48]. Specifically, LoRA allows the reshaping of an embedding space of large-scale models (*i.e.* CLIP in our scenario) by optimizing a small subset of parameters. This effect can be particularly useful in the task of deepfake detection, especially when facing scarcity in data samples, as it can effectively limit the overfitting phenomenon during fine-tuning [52]. In this paper, we conduct an experimental investigation into the few-shot learning capabilities of CLIP-based deepfake detection systems, evaluating their performance against four different state-of-the-art generative models across training sets of 20, 50, 100, and 1000 samples. Moreover, we propose a low-rank adaptation [27] of the CLIP backbone, demonstrating that efficient fine-tuning can consistently outperform other methodologies, starting with 50 pairs of real and fake images. Finally, we test the generalizable capabilities of our proposed methodology when faced with generators unseen during training, finding that LoRA reshapes CLIP embedding space toward generalized detection across different generators.

2 Related Work

Image generation models. Synthetic images are generally created using three different approaches: autoregressive models [18,21,41,57], generative adversarial networks (GANs) [6,11,29,33,50], and diffusion models [2,17,26,35,49]. Our work considers images coming from more than one family of approaches. In fact, the generated data we consider originates from Stable Diffusion [42], both the 1.4 and 2.1 versions, ProGAN [29], and DALL-E 3 [4]. To structure the image distribution, ProGAN starts with an easier task (images at low resolution) and then incrementally improves resolution step-wise while progressively adding new layers to both the generator and discriminator. Differently, Stable Diffusion models represent a specific variant of diffusion models. Indeed, these generative models operate within the latent space [34,42], augmenting efficiency while preserving the final image quality. Within the latent space, the diffusion process is conditioned through cross-attention with the U-Net layers [43]. Lastly, we consider DALL-E 3 [4], a state-of-the-art text-to-image commercial tool. This generator is available through an API and is capable of aligning images closely with the textual inputs, due to the adoption of ChatGPT [37] for prompt expansion.

Deepfake detection. The distinction between real and generated images has been an active area of research, where new classifiers are needed as generation techniques improve. Initially, detectors focused on GAN-based face generators [44,51,55]. Subsequently, with the introduction of diffusion models, detectors rapidly adapted to natural images, expanding the horizons of the face domain [1,3,13,20]. Differently from analyzing RGB data, some approaches [13,22] have utilized frequency analysis, as the generated images show spectral features that differ from real ones. Moreover, a different approach to diffusion models is explored by Wang *et al.* [54], who works on the difference between the input image and the one reconstructed by a pre-trained diffusion model.

Within the domain of deepfake detection, a significant challenge is the adaptation to generators not encountered during training, which tests the ability of the model to generalize. Recent approaches respond to this issue by employing CLIP as a pre-trained backbone from which to extract visual features used for deepfake detection [1,14,36,47]. Notably, these approaches do not use the semantic properties derived from the alignment of text and image during pre-training; rather, they leverage distinctive patterns extracted from the visual backbone. Subsequently, these visual features are utilized by classifiers to execute a binary classification task. Classifiers that have been explored in this context include Support Vector Machines (SVMs) [14], linear classifiers [1,36], and k-NN [36]. While the visual features extracted from the pre-trained CLIP embedding space are not specifically trained for deepfake detection, our approach employs LoRA fine-tuning [27] for remodeling the embedding space of CLIP with a small number of samples, with the final goal of improving deepfake classification.

3 Proposed Method

In this paper, we focus on the task of distinguishing real images (*i.e.* captured via photographic devices) from those completely generated through AI systems. In the existing literature, methodologies tackle this challenge through the creation of extensive datasets, considering thousands of real and fake images. In contrast,

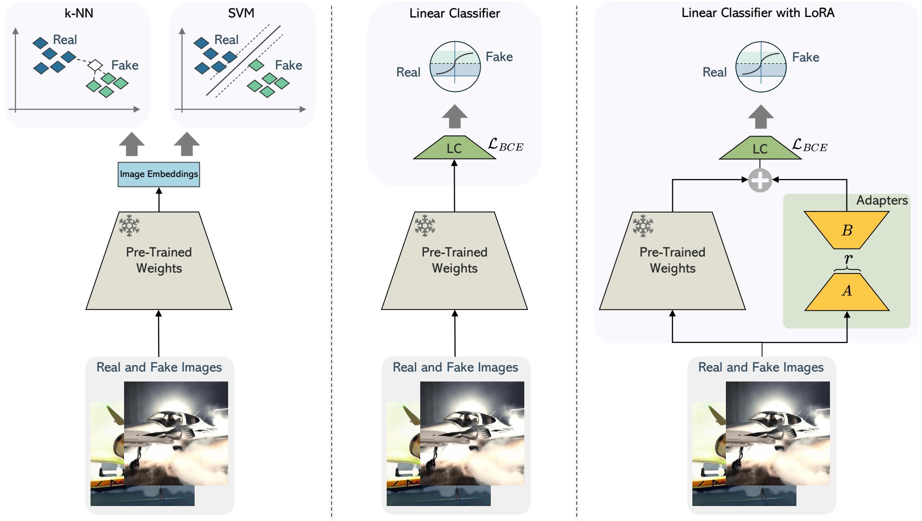

Fig. 1. Illustration of the evaluated deepfake detection classifiers. On the left, images are processed through a pre-trained backbone, with k-NN and SVM classifiers being fitted on the resulting image embeddings. In the center, a linear classifier (LC) is added on top of the backbone and trained using binary cross-entropy. On the right, our proposed fine-tuning protocol using LoRA; where LoRA adapters are added with pre-trained weights and concurrently trained with a linear classifier.

our research explores a distinct scenario where the availability of images from each generator for the training phase is significantly constrained. This assumption is validated in a real-world context wherein a newly released generator is unlikely to publish extensive samples, thereby restricting the availability of data for training purposes. Similarly, for closed-source generators large quantities of images are not publicly available.

3.1 Preliminaries

CLIP architecture [40] has been recently applied in the realm of deepfake detection. Indeed, the visual features extracted from this large-scale model have been proven discriminative in this task, leading to the introduction of multiple binary classifiers (*i.e.* k-NN, SVMs, and linear layers) added on top of the frozen CLIP backbone to perform the task of fake detection [1,14,36].

Employing CLIP for a few-shot classification task offers the advantage of preventing the necessity for initial training. However, the CLIP model was originally developed with a distinct objective, *i.e.* the optimization of image-text similarity. Consequently, the ability of the CLIP embedding space to differentiate between synthetic and authentic images emerges as a secondary function of the architecture, prompting us to adapt the embedding space specifically for the task of deepfake detection.

Given the constraints of few-shot scenarios and building on the hypothesis that features relevant to deepfake detection occupy a compact subspace within the CLIP embedding domain, we investigate the efficacy of LoRA [27] in addressing this issue. In Fig. 1, we represent detectors that leverage image embeddings extracted from a pre-trained backbone. In particular, on the right, the adoption of low-rank adaptation is illustrated.

3.2 LoRA for Deepfake Detection

Given a collection of real images R and fake images F, generated from a specific deepfake generator, we select a small collection $\{(F_i, R_i), i \in (1, N)\}$ of N pairs each composed of a fake image F_i and a real image R_i. Images are firstly cropped to a size of 224^2 and then normalized by a pre-processing pipeline. Secondly, a CLIP visual backbone is employed for feature extraction. Instead of maintaining the weights frozen, we introduce trainable matrices (*i.e.* LoRA adapters), based on rank decomposition and applied into every linear layer of the backbone.

From a mathematical perspective, given a rank r and an initial weight matrix $W_0 \in \mathcal{R}^{d \times k}$, where $r << \min(d, k)$, LoRA introduces a novel formulation for weight matrices as delineated by:

$$W = W_0 + \frac{\alpha}{r} BA. \tag{1}$$

Here, $B \in \mathcal{R}^{d \times r}$ and $A \in \mathcal{R}^{r \times k}$ represent the matrices introduced for adaptation. Throughout the training process, the original weight matrix W_0 remains frozen, while B and A are optimized. Following the original implementation, B is initialized with zeros, and A is initialized from a Gaussian distribution. Conversely, α functions as a hyperparameter, modulating the degree of influence imposed by the matrices introduced by LoRA.

After CLIP processing, each R_i and F_i image is embedded in CLIP embedding space and represented as $\mathcal{F}_{R_i}$ and $\mathcal{F}_{F_i}$ features. A binary linear classifier (LC) is then trained to separate these features into distinct classes through a binary cross-entropy loss. The employment of LoRA adapters facilitates the reshaping of the CLIP embedding space, to separate $\mathcal{F}_R$ and $\mathcal{F}_F$ in the low-rank subspace. A better feature separation would result in an improved classification boundary between real and fake data.

Notably, training images R_i and F_i are chosen to represent the same semantical content. This is done to avoid a real-fake separation inside the CLIP embedding, based on the semantical properties of extracted features.

In Table 1, we detail the number of trainable parameters for each examined LoRA configuration. Significantly, the most extensive configuration encompasses 25M parameters, which corresponds to merely 7% of the Vision Transformer Large model (ViT-L) [19] employed in our experiments. During the evaluation phase, the trainable parameters are combined with the frozen weights of the backbone, as seen in Equation 1. This procedure does not result in an increase in computational load during the inference phase, as the number of parameters remains the same as in the original model.

Table 1. Number of trainable parameters for each examined LoRA configuration and linear classifier (LC) baseline. At evaluation time, adapters and pre-trained weights are merged resulting in the same number of parameters of CLIP LC.

Model	r	α	Trainable Parameters
CLIP LC	-	-	3k
LoRA CLIP LC	16	32	6M
LoRA CLIP LC	32	64	12M
LoRA CLIP LC	64	128	25M

Notably, a reduced rank r reflects in the update of a smaller number of parameters, making it advantageous for training processes that involve limited data. However, this scenario imposes a dimensional limit on the deepfake subspace within the CLIP embedding space; an increased rank may alleviate this limitation. Conversely, fine-tuning the whole visual backbone could face two drawbacks. First, fine-tuning all parameters on a small quantity of data could highly induce the overfitting phenomenon. Second, by completely redefining the CLIP embedding space, it is possible to lose the generalization capability of the network to unseen generators during training.

4 Experiments

In this section, we first describe the evaluation protocol detailing the training data, the backbone used, the baselines employed for the experiments, and the implementation details. Subsequently, we conduct experimental investigations on our proposed LoRA methodology and competitors across various state-of-the-art generative models. Within this context, we consider variations in the number of samples and analyze the generalization capabilities on unseen generators.

4.1 Evaluation Protocol and Experimental Setting

Datasets. The study of the few-shot detection capabilities of deepfake detectors requires an analysis across various deepfake generative methods. This necessity stems from the assumption that different generators may exhibit divergent behaviors in a limited-sample context, thereby requiring a varying quantity of samples to achieve acceptable detection performance. Through our experimentation, we analyze four different state-of-the-art generators, namely ProGAN, Stable Diffusion v1.4, Stable Diffusion v2.1, and DALL-E 3.

In particular, ProGAN [29] represents a popular GAN generator trained on the LSUN dataset [56], which has been deeply analyzed in the context of deepfake detection [36,53]. Differently, Stable Diffusion v1.4 (SD 1.4) and Stable Diffusion v2.1 (SD 2.1) [42] consist of two open-source diffusion models trained on the LAION dataset [46] for text-to-image conditioned generation. Finally, DALL-E 3 [4] represents the latest commercial tool introduced by OpenAI in the field of diffusion models applied to image generation.

We consider a total of 728k images from the collection introduced by Wang *et al.* [53], which includes fake images generated with ProGAN and real images coming from the same LSUN [56] classes as the generated ones. We generate nearly 14k images for both SD 1.4 and SD 2.1. This generation is performed by collecting 14k real images associated with a textual prompt from the LAION-400M dataset [46], which are then used as conditioning text to the diffusion models. Regarding the DALL-E 3 generator, we obtain 10k images from a publicly accessible collection[3]. Given the absence of corresponding real images in the dataset, we combine DALL-E 3 images with randomly selected real images from LAION-400M dataset. From these data sources, we consider 4k and 1k real-fake image pairs, respectively to create the test and validation sets for each of the four considered generators. Moreover, concerning the training set, we sample N pairs of images from the data collection to explore various few-shot scenarios, as will be introduced in Section 4.2.

Given the significant influence of image compression in the context of image forensics [12,24] and acknowledging that the majority of real images from the LAION dataset are encoded in JPEG format, we standardize all images by converting them to JPEG. This ensures uniformity in the dataset, thereby mitigating any potential bias related to varying image compression formats.

Backbone and deepfake detectors. As previously introduced, we primarily focus on the CLIP backbone. Specifically, we employ the CLIP ViT-L model pre-trained on the DataComp dataset [23] and explore different classifiers added on top of the network, such as k-NN, SVM, and linear classifiers.

Following previous literature [36], we implement a k-NN classifier, setting $k = 3$ and employing cosine distance. In this case, a feature bank is constructed by processing the training images and storing the extracted features. During the evaluation phase, the class of an image is determined by identifying the three feature vectors within the bank that exhibit the highest cosine similarity to the feature vector of the given example image. Distinctly, another baseline introduces a Support Vector Machine (SVM) classifier with a linear kernel, adopting the approach proposed by Cozzolino *et al.* [14]. Both k-NN and SVM classification processes are depicted on the left side of Fig. 1. Furthermore, we construct an additional classifier by integrating a linear classifier (LC) for binary classification on top of the CLIP backbone. This deepfake classifier is trained with binary cross-entropy loss, and a threshold of 0.5 is employed for separating real and fake images. Following previous research efforts [20,53], we additionally conduct experiments using a ResNet50 architecture [25] pre-trained on ImageNet and combined with a linear classifier.

Differently, our proposal consists of adding LoRA adapters to all the ViT-L linear layers (*i.e.* multilayer perceptron and attention layers). In our experiments, we apply the adapters only on the weight matrices, excluding the biases, and maintain a constant ratio of α to r, fixed at a value of 2 to balance adaptation and stability. Additionally, our configuration leverages a linear classifier on top of the backbone.

[3] `https://huggingface.co/datasets/OpenDatasets/dalle-3-dataset`

Table 2. Accuracy results when training with 20, 50, and 100 pairs of real and fake images and testing on the same generator. The results represent the average on five different runs with different pairs of images.

	ProGAN			**SD 1.4**			**SD 2.1**			**DALL-E 3**		
Model	20	50	100	20	50	100	20	50	100	20	50	100
ResNet50 LC	50.2	50.5	50.5	51.0	51.6	51.7	50.3	51.1	51.2	52.0	52.0	52.2
CLIP *k*-NN	62.7	65.6	68.0	56.8	57.0	57.7	57.2	58.2	59.2	59.3	63.3	68.6
CLIP SVM	**88.5**	91.6	93.2	**69.8**	73.8	76.1	**70.8**	75.5	76.6	**87.7**	**90.1**	91.5
CLIP LC	85.8	90.8	92.6	68.3	73.7	76.7	68.8	74.9	77.7	82.9	88.4	91.0
LoRA CLIP LC	88.1	**93.2**	**96.0**	69.4	**75.4**	**79.7**	70.2	**76.8**	**79.5**	82.5	89.7	**92.1**

Implementation details. With a limited number of training samples, the use of image transformations emerges as a critical operation to mitigate the risk of overfitting. As a consequence, we select various types of image transformations, including blur, brightness, aspect ratio, pixelization, rotation, contrast, saturation, encoding quality, opacity, overlay stripes, pad, scale, sharpen, skew, grayscale, and horizontal flip. During training, each image is subjected to a stochastic process where the number of transformations applied is randomly selected from a range between 0 and 2. This approach is designed to introduce controlled variability into the training data without visually compromising images by applying too much data transformation. Moreover, to increase the variability of our data, each chosen image transformation is applied with a random strength value. This is sampled from five equally spaced ranges, generated by dividing the interval between a minimum and maximum value that we set for each transformation, with the aim of maintaining visual consistency and usability. With this configuration, we obtain five unique variants for every transformation, each with a different bounded level of intensity. Considering this random selection, all training images undergo random cropping to a dimension of 224^2. Conversely, during the evaluation phase, only a center-crop transformation is applied, at 224^2. Following this pre-processing step, each image is processed by a visual backbone. Specifically, when employing the CLIP, feature extraction is conducted from the next-to-last layer, following [14]. This approach avoids the final linear projection into the shared image-text CLIP embedding space.

From a technical standpoint, model training is performed with batch size 16, a learning rate set to $1e^{-3}$, and the SGD optimizer. The training consists of a maximum of 150 epochs, while the learning rate is reduced by a factor of 10 whenever no validation accuracy improvement is faced in the last 10 epochs. Training is automatically stopped if the learning rate reaches $1e^{-7}$. Considering the limited volume of training samples typically encountered, the evaluation phase is scheduled to occur after every two epochs of training, thereby optimizing computational efficiency.

4.2 Experimental Results

We evaluate the performance of deepfake detectors across a variety of few-shot scenarios. In particular, detectors are trained on varying numbers of pairs of

Table 3. Accuracy results when training with 1000 pairs of real and fake images and testing on the same generator.

	ProGAN	**SD 1.4**	**SD 2.1**	**DALL-E 3**
Model	1000	1000	1000	1000
ResNet50 LC	50.4	52.0	51.4	52.6
CLIP *k*-NN	76.3	62.7	65.9	79.3
CLIP SVM	96.6	82.2	79.7	91.8
CLIP LC	97.4	83.4	84.3	92.8
LoRA CLIP LC	**99.5**	**94.1**	**90.7**	**95.7**

samples (real and fake) N, specifically 20, 50, 100, and 1000. Considering the limited sample size in scenarios where $N \in \{20, 50, 100\}$, we conduct the experiments across five distinct random seeds, reporting the average results. This operation allows the selection of diverse sets of image pairs for each iteration, to maximize the robustness of the experimental configuration. Conversely, in the case where $N = 1000$, the results from a single random seed are reported, given that the increased number of samples inherently guarantees better stability.

Evaluation on few examples. In Table 2, we report the accuracy results of our LoRA-modified CLIP model in comparison to other deepfake classifiers, specifically in scenarios characterized by a limited number of examples, namely $N \in \{20, 50, 100\}$. Notably, the efficacy of the detection models varies across different generative models. For example, employing CLIP with an SVM classifier yields accuracies of 88.5% and 87.7% for ProGAN and DALL-E 3, respectively, with $N = 20$. However, the accuracy diminishes to 69.8% and 70.8% when applied to SD 1.4 and SD 2.1, respectively. Similarly, with $N = 50$, our LoRA-enhanced model achieves accuracies of 75.4% and 76.8% for SD 1.4 and SD 2.1, respectively, whereas the results are notably higher for ProGAN and DALL-E 3, standing at 93.2% and 89.7%. This variance in performance is attributed to the various representations of different generators within the CLIP embedding space, resulting in the importance of evaluating few-shot accuracy across a spectrum of different types of generators.

When analyzing the effectiveness of detection strategies, it is noticeable that the LC paired with ResNet50 underperforms. For instance, this classifier achieves a mere 2.2% improvement in accuracy compared to random choice accuracy, *i.e.* 50%, on DALL-E 3 with $N = 100$. Differently, in the same configuration, CLIP combined with LC obtains an accuracy of 91%. This proves the effectiveness of leveraging large-scale models for few-shot deepfake detection.

Comparing our LoRA detector with the classifiers, it is evident that while performance is comparable with $N = 20$, our proposal obtains the best results with $N = 50$ and $N = 100$. For instance, LoRA CLIP obtains 93.2% and 79.7% with $N = 50$ and $N = 100$ respectively on ProGAN and SD 1.4, obtaining a gain of 1.6% and 3.6% over the SVM mode. Also, our solution demonstrates superior performance compared to the baseline CLIP LC in the majority of comparisons. Notably, even with a smaller sample size $N = 20$, our model surpasses this competitor across ProGAN, SD 1.4, and SD 2.1, with accuracy improvements of

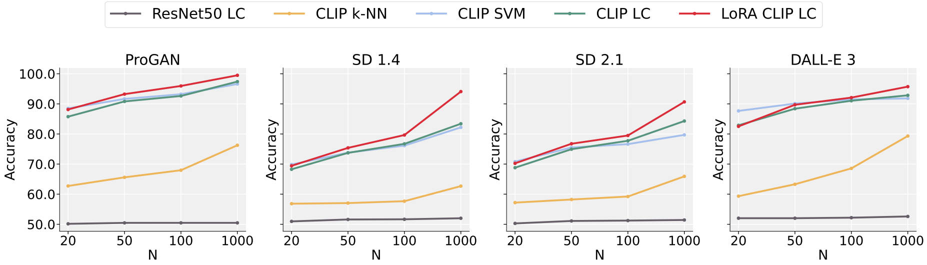

Fig. 2. Trend of accuracy scores on multiple generators. Each classifier is trained on different numbers of samples N, with $N \in \{20, 50, 100, 1000\}$ and tested on the same generator. An accuracy of 0.5 indicates that performance is equivalent to random choice.

2.3%, 1.1%, and 1.4% respectively. This indicates the efficacy of adapting the CLIP embedding space for deepfake detection even with minimal data availability, underscoring the adaptability of our proposal in a few-shot scenario.

Evaluation on more examples. In Table 3, we detail the accuracy scores of detectors, now evaluated in the context of $N = 1000$ sample pairs. Although we still consider this a few-shot scenario, it presents a relaxed constraint compared to the previous analysis.

Our LoRA-enhanced CLIP model surpasses all competitor models across all generators. Remarkably, our approach achieves accuracy improvements of 2.1%, 10.7%, 6.4%, and 2.9% over the CLIP LC model, which is the second most effective in this comparison. Further, CLIP equipped with a linear classifier demonstrates superior scalability in the $N = 1000$ scenario compared to the SVM classifier across all generators, showing performance gains of 0.8%, 1.2%, 4.6%, and 1% for ProGAN, SD 1.4, SD 2.1, and DALL-E 3, respectively.

Finally, while the performance of all methods across all generators tends to increase, as expected, with the increase of N, our proposed model demonstrates a more pronounced improvement in response to the increment of N. This trend is visually delineated in Fig. 2, providing evidence of the scalability of our approach when training size increases.

Effects of hyperparameters r and α on LoRA performance. In Table 4, we report an ablation study on the LoRA hyperparameter rank r when tested on ProGAN and SD 1.4 generators. Notably, the accuracy scores of the LoRA CLIP model show a positive correlation with the hyperparameter r. Specifically, within the context of SD 1.4 and a sample size of $N = 50$, $r = 16$ obtains an accuracy of 73.4%, while using $r = 32$ and $r = 64$ reach an accuracy of 74.9% and 75.4% respectively. Moreover, considering $N = 20$, the $r = 64$ configuration performs better than $r = 16$ on both ProGAN and SD 1.4 with accuracy gains of 2.6% and 0.5%. This performance improvement is particularly remarkable given the substantial increase in learnable parameters, nearly 20M, associated with the $r = 64$ configuration compared to $r = 16$. Moreover, across all configurations of r, LoRA models demonstrate superior performance in comparison to the baseline CLIP LC model, proving the validity of our introduced approach independently by the analyzed hyperparameter choice.

Table 4. Accuracy results of different LoRA configurations when training with 20, 50, 100, and 1000 pairs of real and fake images and testing on the same generator. When considering 20,50, and 100 samples, the results represent the average on five different runs with different pairs of images.

			SD 1.4				ProGAN			
Model	r	α	20	50	100	1000	20	50	100	1000
CLIP LC	-	-	68.3	73.7	76.7	83.4	85.8	90.8	92.6	97.4
FT CLIP LC	-	-	65.3	69.9	77.2	**96.1**	52.6	54.8	60.5	99.1
LoRA CLIP LC	16	32	68.9	73.4	77.5	88.9	85.6	91.1	93.5	99.1
LoRA CLIP LC	32	64	68.3	74.9	78.8	90.9	86.5	92.2	94.7	99.2
LoRA CLIP LC	64	128	**69.4**	**75.4**	**79.7**	94.1	**88.2**	**93.2**	**96.0**	**99.5**

Additionally, we consider a traditional fine-tuned CLIP (FT CLIP LC) where we update all the weights of the backbone for the deepfake task. As highlighted in Table 4, a complete fine-tuning causes poor performance when $N \in \{20, 50, 100\}$. For instance, when training on ProGAN generator we report an accuracy of 52.6%, 54.8%, and 60.5% when training respectively on 20, 50, and 100 couples of real-fake images. This can be attributable to an overfitting scenario caused by a lack of training data. When complete fine-tuning is applied, the performance tends to increase when relaxing the few-shot constraint to $N = 1000$.

Validation on unseen generators. While the primary focus of this paper is on evaluating the efficacy of deepfake detectors in few-shot learning scenarios, our investigation extends to asses how the classifiers perform on images generated by generative models not encountered during their training phase. Specifically, we analyze results on different diffusion models, namely Guided [17], LDM [42], GLIDE [35] and an autoregressive generator in DALL-E [41]. Further, we analyze a selection of GAN-generated images from ProGAN [29], CycleGAN [58], BigGAN [6], StyleGAN [30], GauGAN [38], StarGAN [11], and other generative models, namely Deepfake [44], SITD [9], SAN [15], CRN [10], and IMLE [31].

We report in Table 5 and Table 6 the results of our LoRA CLIP and competitors on images generated by the previously mentioned generative models, following the datasets introduced by Ojha *et al.* [36] and Wang *et al.* [53] respectively. In addition to our baselines, we report the results obtained with the released checkpoints of the CLIP-based linear classifier introduced in [36] and both ResNet50 versions proposed in [53]. It is worth noting that while our proposal and baselines are trained on ProGAN with $N = 1000$, both the introduced competitors are trained on 360k real-fake pairs from ProGAN and LSUN.

Upon analysis of Table 5, LoRA CLIP exhibits superior performance over all baseline models, achieving accuracy improvements of 1.0%, 4.5%, and 11.7% in comparison to CLIP LC, SVM, and k-NN classifiers, respectively. These results underscore the effectiveness of LoRA-adapted embedding space in enhancing detection capabilities on unseen generators, towards a generalized deepfake detection embedding space. Compared to the CLIP linear classifier proposed in [36], our LoRA CLIP obtains comparable results with an average loss on performance of -0.4% but with leveraging 360 times fewer training samples.

Table 5. Accuracy results of detectors trained on ProGAN and tested on external generators [36] unseen during training. The symbol † represents pre-trained models, released by the authors, trained on 320k samples.

		LDM			GLIDE				
Model	Guided	200	200 (CFG)	100	100 (27)	50 (27)	100 (10)	DALL-E	**Avg**
CLIP LC† [36]	69.5	94.4	74	95.0	78.5	79.1	77.9	87.3	82.0
ResNet50 0.1† [53]	62.0	53.9	55.3	55.1	60.3	62.7	61.0	56.1	58.3
ResNet50 0.5† [53]	52.3	51.1	51.4	51.3	53.3	55.6	54.3	52.5	52.7
CLIP *k*-NN	61.3	73.6	67.2	73.9	71.4	72.1	71.3	68.8	69.9
CLIP SVM	63.5	85.4	64.5	87.3	82.0	82.0	81.8	70.2	77.1
CLIP LC	67.4	91.4	64.5	92.7	**87.1**	**86.1**	**85.5**	70.4	80.6
FT CLIP LC	54.3	76.4	67.1	77.1	61.9	62.1	62.9	72.9	66.8
LoRA CLIP LC	**68.4**	**93.9**	**68.3**	**94.4**	83.6	83.5	83.4	**77.3**	**81.6**

Table 6. Accuracy results of detectors trained on ProGAN and tested on external generators [53] unseen during training. The symbol † represents pre-trained models, released by the authors, trained on 320k samples.

	ProGAN	CycleGAN	BigGAN	StyleGAN	GauGAN	StarGAN	DeepFake	SITD	SAN	CRN	IMLE	**Avg**
CLIP LC† [36]	99.8	98.3	95.1	84.9	99.5	95.8	68.6	62.2	56.6	56.6	69.1	80.6
ResNet50 0.1† [53]	100	85.2	70.2	87.1	78.9	91.8	53.5	90.3	50.5	86.3	86.2	80.0
ResNet50 0.5† [53]	100	80.8	59.0	73.4	79.3	81.0	51.1	78.3	50	87.6	94.1	75.9
CLIP *k*-NN	79.6	80.2	68.1	65.9	81.4	72.3	55.0	55.6	59.4	60.1	61.2	67.1
CLIP SVM	98.8	84.9	80.0	77.2	95.1	75.1	**63.4**	70.0	59.4	61.3	61.6	75.1
CLIP LC	99.1	85.9	81.3	77.8	96.9	69.8	61.2	71.1	64.2	61.4	70.2	76.3
FT CLIP LC	99.7	87.7	83.8	78.7	95.9	68.3	54.7	72.8	55.3	61.5	88.6	77.0
LoRA CLIP LC	**99.8**	**95.8**	**91.7**	**85.0**	**99.6**	**77.4**	59.2	**75.0**	**66.0**	**91.8**	**97.6**	**85.3**

Differently, in Table 6 LoRA CLIP obtains the best result on average compared to all the analyzed methodologies. Specifically, our solution outperforms CLIP LC, SVM, and *k*-NN by respectively 9.0%, 10.2%, and 18.2% accuracy on average. Additionally, improvements of 4.7%, 5.3%, and 9.4% are obtained in comparison to the CLIP-based detector proposed in [36] and both detectors proposed by Wang *et al.* [53]. This further provides evidence of the efficacy of LoRA adaptation in the research field of deepfake detection. Furthermore, it is interesting to note that traditional fine-tuning (FT CLIP) loses generalization capabilities on unseen generators reporting a deficit accuracy on average of −14.8% and −8.3% when compared to LoRA CLIP in Table 5 and Table 6 respectively. This is likely due to the overfitting on the ProGAN generator observed during training. Fine-tuning all parameters completely modifies the deepfake subspace inside the CLIP embedding space, thus losing generalization capabilities.

5 Conclusion

In this study, we analyze the efficacy of CLIP-based deepfake detectors under conditions of few-shot learning, assessing their performance across various generators. Moreover, we introduce LoRA CLIP, aimed at refining the CLIP embedding space for the task of deepfake detection. The experimental results validate the effectiveness of our proposed method in identifying synthetic images within few-shot contexts. Further, the LoRA-enhanced CLIP model exhibits significant generalization capabilities to previously not encountered generative models.

Acknowledgments We acknowledge the CINECA award under the ISCRA initiative, for the availability of high-performance computing resources and support. This work has been supported by the Horizon Europe project "European Lighthouse on Safe and Secure AI (ELSA)" (HORIZON-CL4-2021-HUMAN-01-03), co-funded by the European Union.

References

1. Amoroso, R., Morelli, D., Cornia, M., Baraldi, L., Del Bimbo, A., Cucchiara, R.: Parents and Children: Distinguishing Multimodal DeepFakes from Natural Images. ACM TOMM (2024)
2. Balaji, Y., Nah, S., Huang, X., Vahdat, A., Song, J., Zhang, Q., Kreis, K., Aittala, M., Aila, T., Laine, S., et al.: eDiff-I: Text-to-Image Diffusion Models with an Ensemble of Expert Denoisers. arXiv preprint arXiv:2211.01324 (2022)
3. Baraldi, L., Cocchi, F., Cornia, M., Baraldi, L., Nicolosi, A., Cucchiara, R.: Contrasting Deepfakes Diffusion via Contrastive Learning and Global-Local Similarities. In: ECCV (2024)
4. Betker, J., Goh, G., Jing, L., Brooks, T., Wang, J., Li, L., Ouyang, L., Zhuang, J., Lee, J., Guo, Y., et al.: Improving image generation with better captions (2023)
5. Betti, F., Staiano, J., Baraldi, L., Baraldi, L., Cucchiara, R., Sebe, N.: Let's ViCE! Mimicking Human Cognitive Behavior in Image Generation Evaluation. In: ACM Multimedia (2023)
6. Brock, A., Donahue, J., Simonyan, K.: Large Scale GAN Training for High Fidelity Natural Image Synthesis. In: ICLR (2018)
7. Bucciarelli, D., Moratelli, N., Cornia, M., Baraldi, L., Cucchiara, R., et al.: Personalizing Multimodal Large Language Models for Image Captioning: An Experimental Analysis. In: ECCV Workshops (2024)
8. Caffagni, D., Cocchi, F., Barsellotti, L., Moratelli, N., Sarto, S., Baraldi, L., Baraldi, L., Cornia, M., Cucchiara, R.: The Revolution of Multimodal Large Language Models: A Survey. In: ACL Findings (2024)
9. Chen, C., Chen, Q., Xu, J., Koltun, V.: Learning to See in the Dark. In: CVPR (2018)
10. Chen, Q., Koltun, V.: Photographic Image Synthesis with Cascaded Refinement Networks. In: ICCV (2017)
11. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. In: CVPR (2018)
12. Cocchi, F., Baraldi, L., Poppi, S., Cornia, M., Baraldi, L., Cucchiara, R.: Unveiling the Impact of Image Transformations on Deepfake Detection: An Experimental Analysis. In: ICIAP (2023)
13. Corvi, R., Cozzolino, D., Poggi, G., Nagano, K., Verdoliva, L.: Intriguing Properties of Synthetic Images: From Generative Adversarial Networks to Diffusion Models. In: CVPR Workshops (2023)
14. Cozzolino, D., Poggi, G., Corvi, R., Nießner, M., Verdoliva, L.: Raising the Bar of AI-generated Image Detection with CLIP. In: CVPR Workshops (2024)

15. Dai, T., Cai, J., Zhang, Y., Xia, S.T., Zhang, L.: Second-Order Attention Network for Single Image Super-Resolution. In: CVPR (2019)
16. Dettmers, T., Pagnoni, A., Holtzman, A., Zettlemoyer, L.: QLoRA: Efficient Finetuning of Quantized LLMs. In: NeurIPS (2023)
17. Dhariwal, P., Nichol, A.: Diffusion Models Beat GANs on Image Synthesis. In: NeurIPS (2021)
18. Ding, M., Yang, Z., Hong, W., Zheng, W., Zhou, C., Yin, D., Lin, J., Zou, X., Shao, Z., Yang, H., et al.: CogView: Mastering Text-to-Image Generation via Transformers. In: NeurIPS (2021)
19. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
20. Epstein, D.C., Jain, I., Wang, O., Zhang, R.: Online Detection of AI-Generated Images. In: ICCV Workshops (2023)
21. Esser, P., Rombach, R., Ommer, B.: Taming Transformers for High-Resolution Image Synthesis. In: CVPR (2021)
22. Frank, J., Eisenhofer, T., Schönherr, L., Fischer, A., Kolossa, D., Holz, T.: Leveraging frequency analysis for deep fake image recognition. In: ICML (2020)
23. Gadre, S.Y., Ilharco, G., Fang, A., Hayase, J., Smyrnis, G., Nguyen, T., Marten, R., Wortsman, M., Ghosh, D., Zhang, J., et al.: DataComp: In search of the next generation of multimodal datasets. In: NeurIPS (2024)
24. Grommelt, P., Weiss, L., Pfreundt, F.J., Keuper, J.: Fake or JPEG? Revealing Common Biases in Generated Image Detection Datasets. arXiv preprint arXiv:2403.17608 (2024)
25. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In: CVPR (2016)
26. Ho, J., Jain, A., Abbeel, P.: Denoising Diffusion Probabilistic Models. In: NeurIPS (2020)
27. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-Rank Adaptation of Large Language Models. In: ICLR (2022)
28. Hu, Z., Wang, L., Lan, Y., Xu, W., Lim, E.P., Bing, L., Xu, X., Poria, S., Lee, R.K.W.: LLM-Adapters: An Adapter Family for Parameter-Efficient Fine-Tuning of Large Language Models. arXiv preprint arXiv:2304.01933 (2023)
29. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive Growing of GANs for Improved Quality, Stability, and Variation. In: ICLR (2018)
30. Karras, T., Laine, S., Aila, T.: A Style-Based Generator Architecture for Generative Adversarial Networks. In: CVPR (2019)
31. Li, K., Zhang, T., Malik, J.: Diverse Image Synthesis from Semantic Layouts via Conditional IMLE. In: ICCV (2019)
32. Li, Y., Yang, X., Sun, P., Qi, H., Lyu, S.: Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics. In: CVPR (2020)
33. Liao, W., Hu, K., Yang, M.Y., Rosenhahn, B.: Text to Image Generation with Semantic-Spatial Aware GAN. In: CVPR (2022)
34. Ni, H., Shi, C., Li, K., Huang, S.X., Min, M.R.: Conditional image-to-video generation with latent flow diffusion models. In: CVPR (2023)
35. Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., Chen, M.: GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models. In: ICML (2022)
36. Ojha, U., Li, Y., Lee, Y.J.: Towards Universal Fake Image Detectors That Generalize Across Generative Models. In: CVPR (2023)

37. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al.: Training language models to follow instructions with human feedback. NeurIPS (2022)
38. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic Image Synthesis with Spatially-Adaptive Normalization. In: CVPR (2019)
39. Poppi, S., Poppi, T., Cocchi, F., Cornia, M., Baraldi, L., Cucchiara, R.: Safe-CLIP: Removing NSFW Concepts from Vision-and-Language Models. In: ECCV (2024)
40. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning Transferable Visual Models From Natural Language Supervision. In: ICML (2021)
41. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. In: ICML (2021)
42. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022)
43. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: MICCAI (2015)
44. Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., Nießner, M.: Faceforensics++: Learning to detect manipulated facial images. In: ICCV (2019)
45. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. In: NeurIPS (2022)
46. Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., Komatsuzaki, A.: LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs. In: NeurIPS Workshops (2021)
47. Sha, Z., Li, Z., Yu, N., Zhang, Y.: DE-FAKE: Detection and Attribution of Fake Images Generated by Text-to-Image Generation Models. In: ACM CCS (2023)
48. Shah, V., Ruiz, N., Cole, F., Lu, E., Lazebnik, S., Li, Y., Jampani, V.: ZipLoRA: Any Subject in Any Style by Effectively Merging LoRAs. arXiv preprint arXiv:2311.13600 (2023)
49. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep Unsupervised Learning using Nonequilibrium Thermodynamics. In: ICML (2015)
50. Tao, M., Bao, B.K., Tang, H., Xu, C.: GALIP: Generative Adversarial CLIPs for Text-to-Image Synthesis. In: CVPR (2023)
51. Wang, R., Juefei-Xu, F., Ma, L., Xie, X., Huang, Y., et al.: FakeSpotter: A Simple yet Robust Baseline for Spotting AI-Synthesized Fake Faces. In: IJCAI (2020)
52. Wang, S., Chen, L., Jiang, J., Xue, B., Kong, L., Wu, C.: LoRA Meets Dropout under a Unified Framework. arXiv preprint arXiv:2403.00812 (2024)
53. Wang, S.Y., Wang, O., Zhang, R., Owens, A., Efros, A.A.: CNN-generated images are surprisingly easy to spot...for now. In: CVPR (2020)
54. Wang, Z., Bao, J., Zhou, W., Wang, W., Hu, H., Chen, H., Li, H.: DIRE for Diffusion-Generated Image Detection. In: ICCV (2023)
55. Yang, X., Li, Y., Lyu, S.: Exposing deep fakes using inconsistent head poses. In: ICASSP (2019)
56. Yu, F., Zhang, Y., Song, S., Seff, A., Xiao, J.: LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. arXiv preprint arXiv:1506.03365 (2015)
57. Yu, J., Xu, Y., Koh, J.Y., Luong, T., Baid, G., Wang, Z., Vasudevan, V., Ku, A., Yang, Y., Ayan, B.K., et al.: Scaling Autoregressive Models for Content-Rich Text-to-Image Generation. arXiv preprint arXiv:2206.10789 (2022)
58. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In: ICCV (2017)