



# CLIPping the Deception: Adapting Vision-Language Models for Universal Deepfake Detection

Sohail Ahmed Khan  
 University of Bergen  
 Bergen, Norway  
 sohail.khan@uib.no

Duc-Tien Dang-Nguyen  
 University of Bergen  
 Bergen, Norway  
 ductien.dangnguyen@uib.no

## ABSTRACT

The recent advancements in Generative Adversarial Networks (GANs) and the emergence of Diffusion models have significantly streamlined the production of highly realistic and widely accessible synthetic content. As a result, there is a pressing need for effective general purpose detection mechanisms to mitigate the potential risks posed by deepfakes. In this paper, we explore the effectiveness of pre-trained vision-language models (VLMs) when paired with recent adaptation methods for universal deepfake detection. Following previous studies in this domain, we employ only a single dataset (ProGAN) in order to adapt CLIP for deepfake detection. However, in contrast to prior research, which rely solely on the visual part of CLIP while ignoring its textual component, our analysis reveals that retaining the text part is crucial. Consequently, the simple and lightweight Prompt Tuning based adaptation strategy that we employ outperforms the previous SOTA approach by **5.01%** mAP and **6.61%** accuracy while utilizing less than one third of the training data (200k images as compared to 720k). To assess the real-world applicability of our proposed models, we conduct a comprehensive evaluation across various scenarios. This involves rigorous testing on images sourced from **21** distinct datasets, including those generated by GANs-based, diffusion-based and commercial tools. **Code and pre-trained models:** <https://github.com/sohailahmedkhan/CLIPping-the-Deception>

## CCS CONCEPTS

- Computing methodologies → Computer vision.

## KEYWORDS

deepfake detection, transfer learning, vision-language models

### ACM Reference Format:

Sohail Ahmed Khan and Duc-Tien Dang-Nguyen. 2024. CLIPping the Deception: Adapting Vision-Language Models for Universal Deepfake Detection. In *Proceedings of the 2024 International Conference on Multimedia Retrieval (ICMR '24)*, June 10–14, 2024, Phuket, Thailand. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3652583.3658035>

## 1 INTRODUCTION

The internet is now flooded with synthetic images generated by deep neural networks, commonly known as "Deepfakes," thanks to



This work is licensed under a Creative Commons Attribution International 4.0 License.

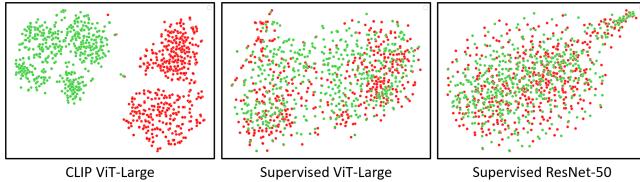
technologies like Generative Adversarial Networks (GANs) [15, 22] and Denoising Diffusion Probabilistic Models (DDPMs) [40, 45]. These powerful tools have become accessible to a wider audience due to open-source availability.

In response to this, researchers have been actively proposing novel methods for automatic detection of synthetic content [5, 7, 33, 46, 48]. However, a major issue with existing deepfake detection models is their limited ability to generalize across different data distributions [5, 30, 33]. Deepfake detection is typically posed as a supervised learning problem, where a deep neural network model is trained to differentiate between authentic (*real*) and manipulated (*fake*) images [33, 46, 53]. However, a significant challenge arises: if the model is exclusively trained on a particular category of fake images, its performance may falter when confronted with novel types of manipulated images, i.e., the generalization dilemma [26].

In [33], Ojha et al. suggest that current detection models might be biased towards identifying certain types of fake images because they focus on easily detectable patterns found in those images. As a result, these models might miss out on the subtle features of real images, treating them as if they do not match the patterns learned from the fake images. In order to overcome this, the authors proposed to conduct classification using models that have trained on diverse range of images during their initial training, i.e., models that are not specifically trained for deepfake detection. They proposed to employ large vision-language models, in particular, CLIP (Contrastive Language-Image Pre-training) [38] as a feature extraction model, and train a linear classification head on top for detecting deepfakes. They also observed that CLIP, even without undergoing specific training for classifying real and fake images, exhibits remarkable capability right from the start in discerning between authentic and fake images. Refer to Figure 1 for details.

In [33], Ojha et al. adapted CLIP for deepfake detection using linear probing, and the results they achieved showed strong generalization capabilities as compared to previous state-of-the-art [46] in detecting deepfakes. However, as highlighted in [27, 51], adapting CLIP through linear probing does not exploit its language component, and only relies on visual features, which can lead to sub-optimal performance. Our hypothesis is that by adapting CLIP using both the visual and text encoders, we can enhance detection performance, leading to a more effective and generalizable strategy for deepfake detection. In order to then verify our hypothesis we now raise this question: "Could combining CLIP's visual and textual capabilities further improve deepfake detection methods?"

In pursuit of an answer, we delve into existing research literature focused on adapting Vision-Language Models (VLMs), specifically CLIP [38], for image classification tasks. For instance, Prompt Tuning [51] by Zhou et al. involves adapting a pre-trained CLIP



**Figure 1: Visualization of real (in red) and a subset of StarGAN [6] generated fake (in green) images utilizing t-SNE in the feature space of three different image encoders. CLIP’s feature space demonstrates superior separation of real and fake images as compared to other two supervised models.**

model using language supervision. This method freezes the large CLIP model, and optimizes a small embedding treated as a prompt. In [14], CLIP-Adapter is introduced, which adds a lightweight linear layer inside the CLIP model. During training, the large CLIP model remains frozen, while the smaller linear layer is optimized. Surprisingly, these promising strategies have not been explored in detecting deepfakes. The primary focus of our study is thus **to determine the most effective transfer learning strategy among various options for large vision-language models in the context of deepfake detection**. Moreover, we also pose questions such as, how various experimental conditions might impact the performance of the adopted strategies. This includes examining their ability to generalize to unseen data, performance when trained with limited real or fake image samples, robustness to different post-processing operations, and the impact of using a restricted amount of data for training.

To answer all these questions, we conduct an empirical analysis of the robustness of CLIP [38] when trained using these strategies, and evaluate resulting models on data originating from varied distributions. Specifically, we take the pre-trained CLIP model, and train it for deepfake detection using four distinct strategies, including (1) Fine-tuning, (2) Linear Probing, (3) Prompt Tuning [51] and (4) training an Adapter Network [14]. Following [46] and [33], we employ ProGAN [21] as our training set. However, in contrast to these studies, we only use 200k images for training as compared to 720k images used by these two studies. We analyze our models on an extensive test set comprising of 21 different GAN-based, diffusion-based and commercial image generators. Our approach achieves high classification performance while using less training data as compared to previous approaches.

Our contributions can be summarised as follows:

- We conduct an extensive empirical investigation into four distinct transfer learning strategies aimed at enhancing the adaptability and robustness of CLIP for deepfake detection, while taking inspiration from recent research on adapting large VLMs.
- Through experimentation, we illustrate that our chosen transfer learning strategies, notably Prompt Tuning, beats the current state-of-the-art [33] by a clear margin.
- We carry out few-shot experiments, illustrating excellent performance of our models even when exposed to only 32 *real/fake* samples from each LSUN object category [50], highlighting the effectiveness of the selected lightweight transfer learning strategies.

- Robustness analysis conducted in the presence of post-processing operations such as JPEG compression and Gaussian blurring.
- Analysis of the impact of training set size, demonstrating that CLIP-based detectors achieve solid performance even when trained using a smaller amount of data (20k real fake images).
- We plan on making the associated code and trained models open-source for the benefit of research community.

This paper is organized as follows. In Section 2 we present a brief description of related works. In Section 3 we introduce the problem background, our proposed deepfake detection workflows, and the datasets that we employ for evaluation of our models. In Section 4 we elaborate in detail about the experiments we carried out for the sake of this study, and discuss the achieved results. Finally in Section 5 we conclude our study.

## 2 RELATED WORKS

### 2.1 Pre-trained Vision-Language Models

Recent advancements in large-scale pre-trained models, which integrate vision and language capabilities, have showcased notable success across a variety of tasks encompassing both images and text [1, 19, 38]. The primary rationale driving the extensive adoption of these models lies in their interesting zero-shot capabilities and robustness to distribution shifts.

Radford et al. proposed Contrastive Language-Image Pre-training (CLIP), a large-scale model that exhibits robust zero-shot performance on several downstream tasks including image classification, optical character recognition, image text retrieval, and multiple other tasks [38]. CLIP was pre-trained on a large scale dataset containing 400 million images, and their associated text captions. CLIP was pre-trained utilizing a contrastive loss, aiming to maximize the similarity between corresponding image and text captions compared to dissimilar pairs. Similar to [38], Jia et al. [19] utilized a large-scale noisy dataset containing one billion image-text pairs to pre-train their model. The model was comprised of dual-encoder architecture, which was tasked to align visual and language representations of image-text pairs through a contrastive loss. They showed that a large enough dataset can compensate for its noise, resulting in state-of-the-art representations even with such a straightforward learning approach.

### 2.2 Transfer Learning

Vision and language models like CLIP [38] and ALIGN [19] offer interesting zero-shot capabilities on several different downstream tasks. Yet, to attain performance levels comparable to state-of-the-art models on these downstream tasks, these models require further fine-tuning on task-specific datasets. For example, even on a simple dataset like MNIST [28], the zero-shot CLIP model (ViT-B/16) which was tested in [27] achieved an accuracy of only 55%.

However, it becomes apparent that fine-tuning full model on downstream dataset affects its robustness to distribution shifts [38, 49]. In response to this challenge, several studies have introduced techniques to fine-tune large vision and language models. In [51] Zhou et al. proposed Context Optimization (CoOp), a fine-tuning strategy to adapt vision-language models similar to CLIP for downstream image classification tasks. CoOp injects learnable vectors to a textual prompt’s context (either at the front, middle or end), which

are optimized during fine-tuning by minimizing the classification loss, whereas, both the vision and text encoders of CLIP are kept frozen. Gao et al. introduced CLIP-Adapter [14], a bottleneck layer designed to learn new features during fine-tuning, while keeping the whole CLIP model frozen.

### 2.3 Fake Image Generation and Detection

Deep learning models for fake image generation have been with us for quite some time. Goodfellow et al. initially introduced Generative Adversarial Networks (GANs), a neural network architecture for unconditional fake image generation [15]. Seminal works were targeted on for example, improved training process of GANs [17, 22, 43], improving quality and diversity of the generated images [21, 25] and conditional image synthesis [32, 47].

In more recent times, text-to-image generation models have attracted interest following the introduction of diffusion models [11, 31]. Most of the recent diffusion based image synthesis models, including Stable Diffusion [40], SDXL [37], DALL-E [39], Imagen [42] have demonstrated the ability to produce high quality images. Diffusion models also demonstrate the ability to generate images spanning a diverse range of categories and scenes as compared to GANs.

With the rise of powerful open-source image synthesis models, detecting *fake* images has become increasingly critical. Previous methods trained deep neural network classifiers to distinguish *real* from *fake* images from the same generative model [41]. However, these classifiers often struggle to generalize to *fake* images from different distributions, as noted in studies [26, 53].

Wang et al. [46] proposed a simple yet effective solution to the challenge of detecting images generated by GANs. By training a well-known CNN architecture, ResNet-50 [18], on a single GAN-generated dataset (ProGAN [21]), along with augmentations like JPEG compression and blurring, they significantly improved the model's robustness. This approach performed well even on images generated by different GAN models. Building on this, Gragnaniello et al. [16] modified ResNet-50 for GAN image detection. They avoided down-sampling in initial layers in order to preserve high frequency GAN realted fingerprints, and applied intense augmentations during training, outperforming previous method [46]. Corvi et al. [8] extended work proposed in [16], training the same modified ResNet-50 on the dataset from [46]. They found their model excelled on GAN images but struggled with diffusion models. However, training on images from LDMs [40] yielded success on images generated using diffusion models but not on GAN images.

In a recent study, Ojha et al. [33] noted that previous techniques [46] fail on diffusion model-generated images when initially trained on images generated by GAN models. They utilized a fixed CLIP encoder to train a linear classifier on CLIP features, achieving SOTA results for both GAN and diffusion model-generated images by just training their model on GAN generated images same as [16, 46].

## 3 METHODOLOGY

### 3.1 Background

The ultimate objective of a deepfake detection system is to determine if any given image is (a) authentic: captured using a camera,

or (b) fake: synthesized using a generative model (GAN or diffusion). In this section, we outline the methodologies examined in this study for training our detection model, along with the datasets used to train and evaluate our model. However, we begin by first presenting the baseline [47] and current SOTA [33] approaches proposed recently to address this task. These studies effectively leads us towards our proposed solution.

Wang et al. in [46] trained a ResNet-50 [18] using cross-entropy loss to perform binary classification between *real* and *fake* images, using data they generated using the ProGAN model [21] after training it on 20 different object categories taken originally from LSUN [50]. For each of the 20 object categories, the authors generated 18k synthetic images, totaling up to 360k *fake* images. They incorporated *real* images from the LSUN dataset, amounting to 18k *real* images for each of the 20 object categories. Consequently, their training dataset contained 720k *real* and *fake* images. They demonstrated through comprehensive evaluation that a simple CNN, when trained with meticulous data augmentation techniques like compression and blurring, exhibits effective generalization for deepfake detection on previously unseen data.

Following this, Ojha et al. [33] found the poor performance of [46], when tested on images synthesized by diffusion models. For instance, on images generated by models like Latent [40] and Guided [11] Diffusion models, the detection model's classification accuracy experiences a significant decline, reaching close to chance performance. This implies that during training, the model emphasizes solely on detecting the presence or absence of model specific artifacts in an image, while overlooking other subtle distinguishing features between *real* and *fake* images.

Ojha et al. [33] suggested using a feature space for classification that isn't initially trained towards distinguishing between *real* and *fake* images, aiming to prevent disproportionate recognition of specific artifacts present in fake images. To validate their hypothesis, the authors used CLIP's [38] image encoder (ViT-Large) as a feature extractor, and trained a simple linear model on top. They used the same dataset as in [46] for training. The obtained results supported their hypothesis: their simple approach achieved state-of-the-art performance on previously unseen images from both GAN and diffusion models [33].

While [33] achieves excellent results on most datasets, it still seems to struggle on some datasets, including Guided Diffusion [11], LDM [40], Deepfakes [41], FaceSwap [41], and commercial generators such as DALL-E 3<sup>1</sup>, Adobe Firefly<sup>2</sup> and Midjourney<sup>3</sup>.

### 3.2 Transfer Learning

When applied to adapt vision-language models for downstream vision tasks, linear probing faces a significant drawback as it completely overlooks the language component. As noted in [51], a linear layer trained on visual features serves as a static set of weights exclusively representing visual concepts. Consequently, the semantics embedded in texts remain largely unexplored, and irrelevant during this process. This limitation is exemplified in [33], where only the visual component of CLIP is utilized for deepfake detection, while

<sup>1</sup><https://openai.com/dall-e-3>

<sup>2</sup><https://www.adobe.com/products/firefly.html>

<sup>3</sup><https://www.midjourney.com/>

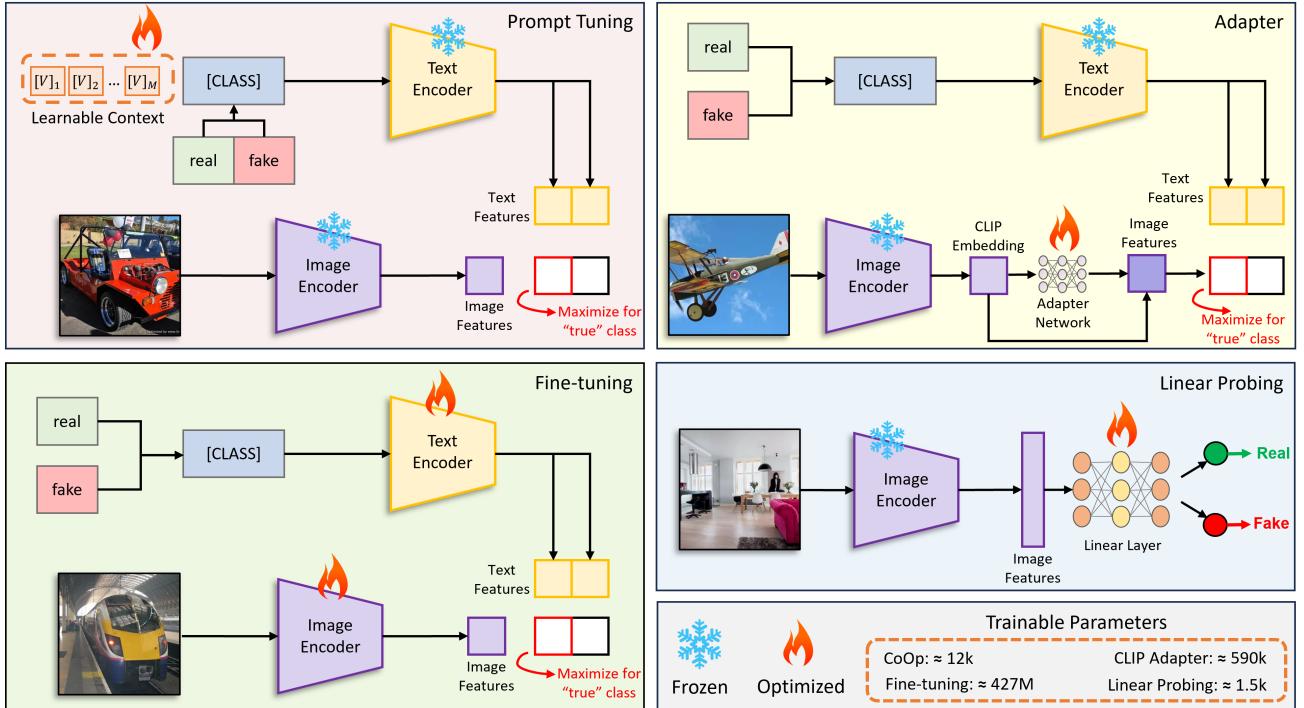


Figure 2: In this figure, we present four distinct transfer learning strategies that are explored for *real/fake* image classification. At bottom right we list the number of trainable parameters for each approach. Images from [13, 50].

completely neglecting the text encoder. We believe that leveraging both the visual and text encoders of CLIP [38] can lead to an improved strategy for *real* vs *fake* classification.

Based on this insight, we propose leveraging CoOp [51], a Prompt Tuning strategy as our central approach to adapt CLIP [38] for deepfake detection. Prompt tuning is particularly appealing as it integrates both the visual and language aspects of CLIP. We employ CLIP-ViT-Large model initially pre-trained on WebImageText (WIT) dataset [38] collected by OpenAI.

To ensure a fair assessment of the robustness of various transfer learning strategies, we incorporate three additional methods, in addition to Prompt Tuning for this task, including (1) Linear Probing, (2) Full Fine-tuning and (3) training an Adapter Network [14]. A concise overview of each employed transfer learning strategy is presented in the following sections.

**3.2.1 Linear Probing:** Linear probing, a well-known transfer learning strategy, involves fine-tuning a linear classifier on top of a frozen model (CLIP in our case). We follow the same approach as employed by Ojha et al. [33], i.e., we discard CLIP’s text encoder while freezing its image encoder. We then train a single linear layer for classification on the frozen CLIP’s image features, mapping the penultimate image features to logits for class predictions using the Sigmoid activation function. The optimization takes place using the binary cross entropy loss. For linear probing and full fine-tuning we employ basic image augmentations including, random flips and cut-outs among some other affine transformations. We illustrate linear probing strategy in Figure 2.

**3.2.2 Fine-tuning:** Fine-tuning in this context means training the whole CLIP model (ViT-Large) again on the downstream dataset, which in our case is the ProGAN dataset which was also used by [46] and [33]. Full fine-tuning requires significantly more compute resources, data, and training time since the entire model is retrained. Additionally, as model size increases, this strategy demonstrates instability and inefficiency [27]. During the training of our models, we encountered this issue, and mitigated it by utilizing an extremely small learning rate,  $1 \times 10^{-6}$ . To fine-tune our model, we adhere to the procedure outlined in the pre-training of CLIP [38]. However, we introduce a modification: rather than utilizing entire text captions for each image, we provide only single-word captions, specifically either *real* or *fake*. A typical Fine-tuning pipeline for adapting CLIP is illustrated in Figure 2.

**3.2.3 Prompt Tuning:** Initially introduced in the domain of natural language processing [29], Prompt Tuning is a relatively recent transfer learning strategy adopted by the computer vision community. This approach involves fine-tuning a pre-trained model like CLIP [38] by learning randomly initialized prompts (textual [51] and/or visual [20]) during training. The primary goal of Prompt Tuning is to adapt the model on specific downstream tasks by optimizing the prompts to align better with the target objectives.

In this study, we employ Context Optimization (CoOp), a transfer learning strategy introduced by Zhou et al. in [51], to fine-tune CLIP for the task of deepfake image detection. CoOp appends learnable vectors along with the context words<sup>4</sup> of a prompt. These learnable

<sup>4</sup>Context words refer to labels of any given dataset. In our case, the labels are *real* and *fake*.

**Table 1:** This table showcases the statistics of the test datasets. Certain datasets include their own collection of *real* images. However, for datasets that lack their own *real* images, we utilize LAION’s [44] images instead.

Generator	Num. <i>real/fake</i>	Real Data Source	Image Resolution	Family
ProGAN [21]	4k / 4k	LSUN	256 x 256	GAN
BigGAN [3]	2k / 2k	ImageNet	256 x 256	GAN
CycleGAN [52]	1k / 1k	Various	256 x 256	GAN
EG3D [4]	1k / 1k	LAION	512 x 512	GAN
GauGAN [36]	5k / 5k	COCO	256 x 256	GAN
StarGAN [6]	2k / 2k	CelebA	256 x 256	GAN
StyleGAN [24]	1k / 1k	LSUN	256 x 256	GAN
StyleGAN2 [25]	1k / 1k	Various	≈ 256 x 256	GAN
StyleGAN3 [23]	≈ 1k / 1k	Various	512 x 512	GAN
Taming-T [12]	1k / 1k	LAION	256 x 256	GAN
DALL-E (mini) [10]	1k / 1k	LAION	256 x 256	-
Glide [31]	1k / 1k	LAION	256 x 256	Diff.
Guided [34]	1k / 1k	LAION	256 x 256	Diff.
LDM [40]	1k / 1k	LAION	256 x 256	Diff.
Stable Diff. [40]	1k / 1k	LAION	512 x 512	Diff.
SDXL [37]	1k / 1k	LAION	1024 x 1024	Diff.
Deepfakes [41]	≈ 2.7k / 2.7k	YouTube	≈ 256 x 256	-
FaceSwap [41]	2.8k / 2.8k	YouTube	≈ 256 x 256	-
Midjourney-V5	1k / 1k	LAION	Various	Comm.
Adobe Firefly	1k / 1k	LAION	Various	Comm.
DALL-E 3	1k / 1k	LAION	Various	Comm.

vectors can be either initialized with random values or pre-trained word embeddings [51]. During training the learnable vectors are optimized whereas the whole CLIP model is kept frozen.

$$t = [V]_1[V]_2 \dots [V]_M[CLASS] \quad (1)$$

Where each  $[V]_m$  ( $m \in 1, \dots, M$ ) is a vector with the same dimension as word embeddings, e.g., 768 for CLIP (ViT-Large) [51].  $M$  is a hyperparameter referring to the number of context tokens, i.e.,  $[V]_M$ . We experiment with  $M = [4, 8, 16, 24]$ .  $[CLASS]$  refers to class token of the dataset, e.g., *real* and *fake* in our case. Class token within each prompt  $t_i$  is swapped with the corresponding word embedding vector of the  $i$ -th class name. The prompt  $t$  is then fed through the text encoder, and optimized using cross-entropy loss during training. As evident from Eq. 1, the context tokens are added at the beginning of the class labels. While the CoOp paper explores various appending strategies, such as "end" and "middle", our findings indicate that appending context tokens at the "front" yields comparably better results. For reference, see Figure 2.

**3.2.4 Adapter Network:** Stepping away from Prompt Tuning, Gao et al. introduced a simple yet effective alternative approach for fine-tuning vision-language models using feature adapters [14]. Specifically, the authors introduce CLIP-Adapter, an extra light-weight bottleneck layer which is optimized during training while the remainder of the CLIP model is kept frozen. Additionally, to remain robust against unseen data distributions, CLIP-Adapter integrates the original zero-shot visual or language embeddings with the corresponding fine-tuning feature embeddings through a ResNet styled residual connection [18]. This feature blending allows CLIP-Adapter to exploit both the knowledge stored in the original CLIP’s feature space, and the newly acquired knowledge from the downstream training examples simultaneously. CLIP-Adapter can

be applied to either the visual branch, language branch or both. In our study however, we only use Adapter Network with Vision branch, and leave the language branch as is. See Figure 2 for reference.

### 3.3 Generative Models Explored

In this paper, we conduct an in-depth investigation into four distinct transfer learning approaches for deepfake detection. Our analysis is aimed at assessing the robustness of these approaches when coupled with pre-trained CLIP [38] ViT-Large model for deepfake detection when exposed to unseen data coming from diverse deepfake generators including GANs and diffusion models.

We follow the same protocols outlined by Wang et al. [46] and Ojha et al. [33], and train our models using data coming from just one generative model i.e., ProGAN [21]. However, for evaluation we incorporate an even broader spectrum of generative models in our analysis. This extension aims to align our evaluation more closely with real-world scenarios. In total, we assess our models across 21 distinct datasets, primarily categorized as GAN-based, diffusion-based and commercial tools [8]. For detailed dataset statistics, please refer to Table 1.

In our research, we utilize GAN and LDM images from [33], along with our self-generated SD [40] and SDXL [37] images. We only generate images containing entities from the 20 categories of the ProGAN dataset [13], like airplanes, cars, and horses.

Another minor fluctuation in evaluation protocol we follow is that [33] employed three distinct configurations for image generation using Glide and LDMs, presenting their findings separately. In contrast, we include all images from Glide and LDM subsets in our analysis but display averaged results in our tables due to space constraints.

## 4 EXPERIMENTS

In this section, we present performance scores achieved by CLIP ViT-Large [38] when trained using four distinct transfer learning strategies: (1) Linear Probing, (2) Fine-tuning, (3) Adapter Network [14] and (4) Prompt Tuning [51]. Additionally, we evaluate trained models released by [8, 16, 33, 46] on the same test set on which we evaluate our own models. Our aim is to determine if our chosen transfer learning strategies offer superior generalization compared to previous studies. In subsequent sections, besides assessing generalization capabilities, we conduct further experiments to assess performance of our models under various conditions, including smaller training set sizes, few-shot analysis, and robustness to post-processing operations.

### 4.1 Generalization Performance

We evaluate our model’s performance by comparing it with four prior studies that aim to detect various types of deepfake images generated by different fake image generators. The initial study [46] in this field employed ResNet-50 [18] as the classifier. They trained their models on 720k *real/fake* images sourced from the ProGAN dataset which they generated for the sake of their study. They also employed image augmentations such as JPEG noise and Gaussian blurring, which made their models more robust towards post-processed images during evaluation. The second study [16] also

**Table 2: Generalization performance.** This table presents the average precision (AP) of different methods for distinguishing *real* and *fake* images. The studied adaptation approaches demonstrate significant improvements over the previous baselines and SOTA.

Method	Variant	Generative Adversarial Networks									DALL-E					Denoising Diffusion Models					FF++		mAP
		Pro GAN	Big GAN	Cycle GAN	EG3D	Gau GAN	Star GAN	Style GAN	Style GAN-2	Style GAN-3	Taming-T	Glide	Guided	LDM	SD	SDXL	Deep Fakes	Face Swap					
Wang et al. (CVPR'20)	Blur+JPEG (0.1)	<b>100.00</b>	83.04	90.09	95.58	88.94	97.18	99.27	96.43	98.63	73.90	67.47	81.02	83.10	68.61	64.33	72.27	75.88	50.78	81.18			
	Blur+JPEG (0.5)	<b>100.00</b>	82.63	94.71	55.32	96.62	93.88	93.25	88.64	85.33	59.78	60.92	69.75	65.11	60.24	52.14	65.92	64.33	49.76	72.65			
Gragn. et al. (ICME'21)	ResNet-50 No Downsample	<b>100.00</b>	97.57	97.63	99.95	98.36	99.99	<b>100.00</b>	<b>99.98</b>	<b>100.00</b>	95.31	91.32	94.08	93.81	92.33	91.75	90.93	<b>95.90</b>	61.54	94.24			
Corvi et al. (ICASSP'23)	ProGAN/LSUN Latent/LSUN	<b>100.00</b>	<b>99.66</b>	97.94	99.92	99.74	99.95	<b>100.00</b>	<b>99.96</b>	99.93	94.34	95.45	89.51	79.30	88.26	87.01	74.90	95.52	56.58	91.52			
Ojha et al. (CVPR'23)	CLIP Linear Probing	99.99	98.73	98.92	79.58	99.74	96.06	95.73	95.81	92.21	97.12	96.84	93.85	92.09	95.71	93.58	88.55	77.48	75.87	93.05			
Ours	Linear Probing	99.91	97.77	98.53	99.48	99.69	99.00	95.53	94.98	99.54	97.74	95.65	97.75	92.14	95.94	92.24	94.99	80.07	76.58	95.22			
	Fine Tuning	<b>100.00</b>	98.65	99.00	99.97	98.12	<b>100.00</b>	99.61	99.48	<b>100.00</b>	98.58	98.15	96.23	97.40	98.79	97.53	<b>99.52</b>	87.42	60.22	96.29			
	Adapter	<b>100.00</b>	<b>99.58</b>	<b>99.97</b>	99.50	<b>99.98</b>	99.98	99.44	98.80	99.83	99.27	98.60	99.26	96.16	97.76	91.90	92.32	91.37	82.11	97.27			
	Prompt Tuning	<b>100.00</b>	99.42	99.92	99.51	99.95	99.97	99.52	98.62	99.68	<b>99.54</b>	<b>98.89</b>	<b>99.32</b>	<b>97.41</b>	97.91	96.23	96.42	92.59	<b>88.01</b>	<b>98.06</b>			

**Table 3: Generalization performance.** This table compares the accuracy (Acc) scores attained by our proposed techniques with various previous studies. The proposed CLIP adaptation strategies show noteworthy performance gains compared to previous baselines and SOTA techniques.

Method	Variant	Generative Adversarial Networks									DALL-E					Denoising Diffusion Models					FF++		Avg. Acc
		Pro GAN	Big GAN	Cycle GAN	EG3D	Gau GAN	Star GAN	Style GAN	Style GAN-2	Style GAN-3	Taming-T	Glide	Guided	LDM	SD	SDXL	Deep Fakes	Face Swap					
Wang et al. (CVPR'20)	Blur+JPEG (0.1)	99.90	67.65	79.50	72.65	76.63	89.72	82.10	77.05	80.68	56.45	55.05	61.15	62.90	54.03	52.50	53.40	52.67	49.68	66.09			
	Blur+JPEG (0.5)	99.65	58.13	77.80	50.30	75.56	79.99	69.80	62.30	53.42	51.05	51.90	54.33	52.35	51.35	50.15	51.00	51.46	50.02	59.18			
Gragn. et al. (ICME'21)	ResNet-50 No Downsample	<b>100.00</b>	93.27	91.75	97.55	94.13	99.65	97.25	89.75	97.47	67.45	60.65	69.38	67.30	62.33	59.70	57.75	65.31	50.02	76.59			
Corvi et al. (ICASSP'23)	ProGAN/LSUN Latent/LSUN	<b>100.00</b>	<b>95.85</b>	90.35	<b>98.40</b>	92.46	99.00	<b>97.65</b>	84.90	82.79	65.30	69.30	58.98	53.10	58.83	55.70	52.10	59.38	50.11	72.72			
Ojha et al. (CVPR'23)	CLIP Linear Probing	98.94	94.48	94.20	57.75	94.65	87.49	85.55	83.40	75.42	89.45	89.20	82.15	79.00	87.80	81.90	74.15	62.71	64.30	82.84			
Ours	Linear Probing	98.50	91.75	91.00	98.20	88.08	94.42	81.40	71.70	94.11	91.05	<b>85.80</b>	90.55	79.05	87.42	77.30	83.85	69.37	68.30	86.26			
	Fine Tuning	99.60	77.38	71.55	<b>98.40</b>	65.70	<b>100.00</b>	94.85	<b>95.30</b>	<b>99.89</b>	94.40	<b>93.20</b>	88.78	<b>92.35</b>	95.17	91.75	<b>97.35</b>	76.46	52.11	88.74			
	Adapter	99.88	94.75	<b>97.45</b>	95.30	<b>95.47</b>	99.12	93.35	78.35	93.11	94.55	92.00	<b>94.27</b>	81.65	89.18	67.70	71.60	77.11	70.16	88.72			
	Prompt Tuning	99.83	93.80	95.60	93.50	93.43	99.15	95.25	82.95	93.11	<b>94.95</b>	91.50	92.88	84.3	88.16	76.45	77.80	<b>78.45</b>	<b>74.66</b>	<b>89.45</b>			

employs ResNet-50, but with a simple adjustment to the original architecture to better preserve the low-level forensic traces present inside images. The proposed modified model was also trained on the ProGAN dataset for *real/fake* classification introduced in [46]. In [8] use the same modified ResNet-50 [16] but train it again on two different datasets, i.e., ProGAN/LSUN and LatentDiffusion/LSUN to better understand which generative model offers better generalization. The fourth study from Ojha et al. [33] attained state-of-the-art performance. They utilized the CLIP ViT-Large model as a feature extractor, and subsequently trained a linear network on top of it for *real/fake* classification.

In Tables 2 and 3, we compare our models' performance with that of [8, 16, 33, 46]. These studies [16, 46] demonstrate strong performance on GAN-generated images but show mediocre results on images from diffusion-based and commercial generators. Conversely, [33] achieves good results on both GAN-based and diffusion-based generators, although performance decreases on images from commercial image generators (see Table 6) and Face-Forensics++ dataset [41], which utilizes an Auto-encoder based architecture for image synthesis.

The presented adaptation approaches for deepfake detection demonstrate consistently better performance across all datasets as apparent from numbers in Tables 2, 3 and 6. However, as seen in Tables 2 and 3, the Prompt Tuning [51] notably outperforms other transfer learning strategies in terms of both mAP and average accuracy. Notably, Prompt Tuning optimizes only a fraction of

parameters (**12k**) compared to the Adapter Network and full Fine-tuning approaches, which optimize a larger number of parameters. Overall, we surpass the previous SOTA [33] by **5.01%** in mAP and **6.61%** in average accuracy across images from **18** distinct synthetic image generators.

## 4.2 Effect of Transfer Learning Strategy

In this section, we assess and compare the effectiveness of transfer learning strategies trained on images from ProGAN/LSUN datasets. Results are summarized in Tables 2 and 3. It is evident from the reported numbers that Prompt Tuning (CoOp) outperforms other strategies. Despite a modest margin, this is noteworthy as Prompt Tuning optimizes only a fraction of parameters ( $\approx 12k$ ) compared to Linear Probing, Fine-tuning and Adapter Network, which optimize approximately ( $\approx 1.5k$ ), ( $\approx 427M$ ) and ( $\approx 590k$ ) parameters respectively. Moreover, in few-shot experiments as shown in Table 5 Prompt Tuning also outperforms other three strategies. However, in terms of robustness to post-processing operations, Linear Probing turns out to be best performing strategy.

## 4.3 Effect of Training Set Size

We also conducted experiments with various training set sizes, and in this section, we report on the performance of participating transfer learning strategies when trained with reduced numbers of *real* and *fake* images. Using ProGAN's [21] data, we create four smaller datasets containing 20k, 40k, 60k and 80k images. As shown in Table 4, we observe that while larger training datasets

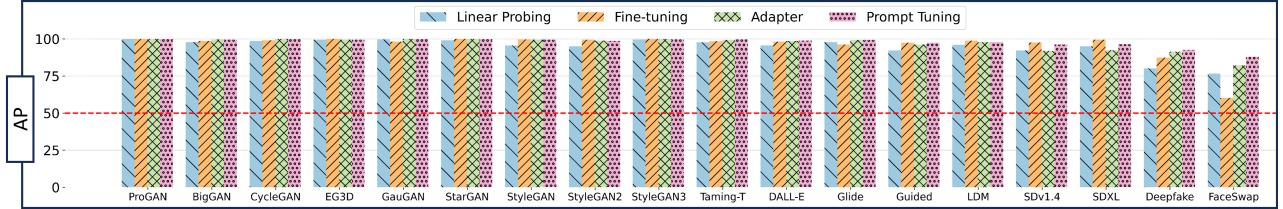


Figure 3: Average precision (AP) score distribution of participating transfer learning strategies on the test set comprised of images sourced from 18 different datasets, as given in Tables 2 and 3. The red dotted line represents chance performance.

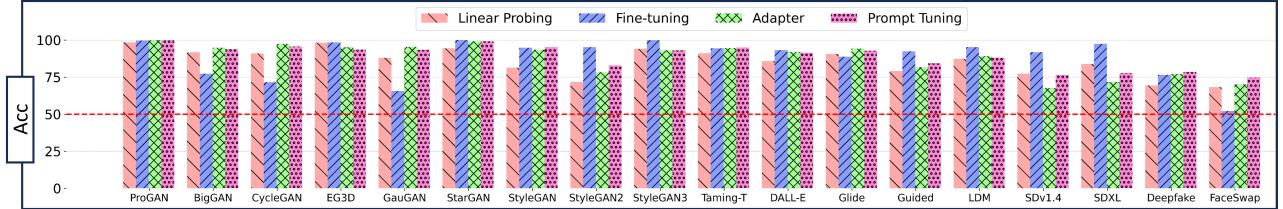


Figure 4: Accuracy (Acc) scores achieved by participating transfer learning strategies on the test set comprised of images sourced from 18 different datasets, as given in Tables 2 and 3. The red dotted line represents chance performance.

generally yield higher scores, the differences are not significant. Moreover, since the *fake* images in the training data are generated by a GAN model (ProGAN [21]), the impact of training data size is less pronounced when evaluating models on other GAN models in the test set compared to diffusion models, or commercial tools. This analysis indicates that even with limited training resources, it is still possible to train robust detection models without a significant decline in generalization capabilities.

#### 4.4 Robustness to Post-processing Operations

In real-world scenarios, images commonly undergo post-processing before being shared online, and research indicates that these operations significantly impact detection models' performance [9, 33, 46]. To assess how our models handle post-processing, following previous studies [33, 46], we evaluate them on images subjected to two types of operations: (1) JPEG compression and (2) Gaussian blurring.

To gauge the impact of JPEG compression, we tested two qualities: 75% and 50%. For blurring effects, we used sigma values of 1 and 2. The performance results of our models are depicted in Figure 5. As expected, there is a decline in performance as sigma and compression values increase, though still acceptable considering our models weren't explicitly trained on compressed or blurred images. One thing we notice is that this decline is more pronounced for images generated by commercial tools, except for fully Fine-tuned model. Linear Probing outperforms other adaptation strategies well across the three different generative model families.

#### 4.5 Few-shot Analysis

We now conduct experiments to investigate how participating transfer learning approaches perform when trained on extremely limited data, specifically only 640 images (320 *real*, 320 *fake*). Here, we present the results achieved by our models in a few-shot setting.

We train CLIP (ViT-Large) model using four different transfer learning strategies, i.e., (1) Linear Probing, (2) Fine-tuning, (3)

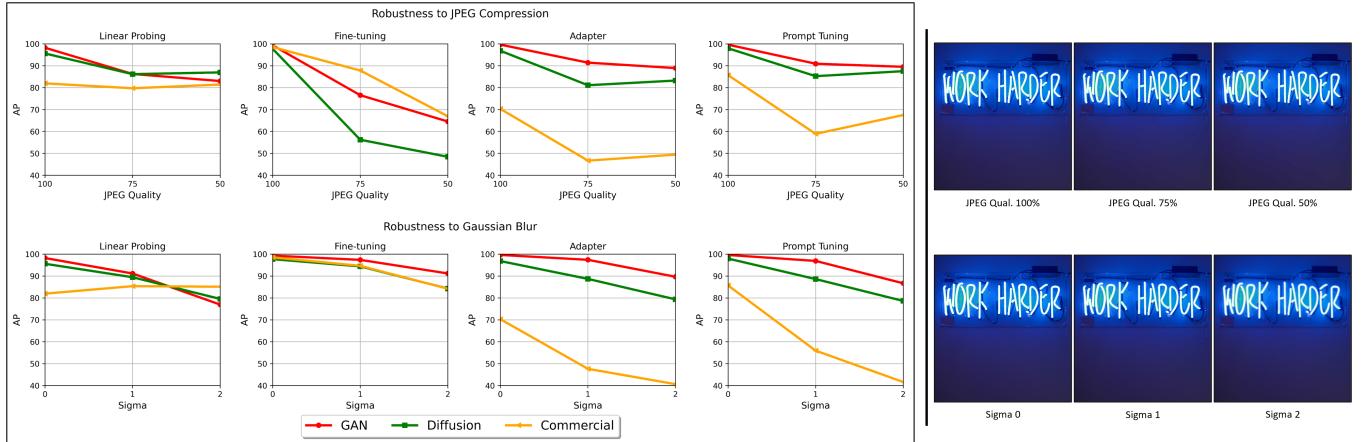
Adapter Network [14] and (4) Prompt Tuning [51] in a few-shot setting. We use only 32 (16 *real* and 16 *fake*) images from each of the object categories available in the LSUN [50] and ProGAN [21] datasets. In total, we train the models using 640 *real/fake* images. We present the achieved Average Precision (AP) and Accuracy (Acc) scores in Table 5. It is apparent from the results that Prompt Tuning outperforms other transfer learning strategies by a clear margin.

#### 4.6 Performance on Commercial Tools

Besides evaluating the models on images sampled by a number of different GAN-based and diffusion-based image generators, following [9] we also carry out evaluations of baseline methods, and the transfer learning strategies we employ on images generated

Table 4: This table presents scores achieved by our models trained using smaller sized datasets. Results are organized based on number of available training images: 20k, 40k, 60k and 80k. We keep equal amount of *real/fake* images, e.g., for 20k subset, we have 10k *real* and 10k *fake* images.

Method	Num. Train Images	Family of Generators			Average AP/Acc
		GAN AP/Acc	Diffusion AP/Acc	Comm. Tools AP/Acc	
Linear Probing		98.86 / 90.78	97.13 / 90.76	80.38 / 74.93	92.12 / 85.49
Fine-tuning	20k	95.68 / 78.10	86.79 / 69.39	71.45 / 63.78	84.64 / 70.43
Adapter		98.57 / 89.17	93.51 / 83.81	62.01 / 53.60	84.70 / 75.53
Prompt Tuning		98.95 / 90.42	96.14 / 87.33	76.10 / 59.62	90.40 / 79.12
Linear Probing		98.94 / 91.28	97.23 / 90.60	77.60 / 73.42	91.26 / 85.10
Fine-tuning	40k	96.83 / 80.10	88.44 / 70.32	69.85 / 63.00	85.04 / 71.14
Adapter		98.98 / 89.69	94.71 / 83.39	62.03 / 52.82	85.24 / 75.30
Prompt Tuning		99.00 / 91.43	96.15 / 86.79	79.52 / 60.10	91.56 / 79.44
Linear Probing		98.97 / 91.33	97.41 / 91.18	77.41 / 73.90	91.26 / 85.47
Fine-tuning	60k	97.08 / 79.91	89.05 / 69.36	70.59 / 62.55	85.58 / 70.60
Adapter		99.35 / 91.74	95.93 / 85.07	64.85 / 53.75	86.71 / 76.85
Prompt Tuning		99.29 / 91.69	96.47 / 85.64	76.94 / 57.25	90.90 / 78.20
Linear Probing		98.94 / 91.38	97.31 / 90.68	76.69 / 73.13	90.98 / 85.06
Fine-tuning	80k	97.75 / 82.79	90.45 / 73.69	72.08 / 65.45	86.76 / 73.98
Adapter		99.46 / 92.12	96.12 / 83.73	66.28 / 53.28	87.29 / 76.38
Prompt Tuning		98.93 / 89.30	96.58 / 85.60	80.95 / 58.08	92.15 / 77.66



**Figure 5:** This figure shows how different transfer learning strategies cope with post-processing operations including JPEG compression and Gaussian blurring. Our models perform well with GAN and diffusion images but struggle with those from commercial tools like DALL-E 3 and Adobe FireFly. Surprisingly, the Fine-tuned CLIP is more robust against compressed images generated using commercial tools as compared to GAN and diffusion-based images. Linear Probing achieves optimal performance across all three datasets. Photo credits: [unsplash.com](https://unsplash.com)

by commercial tools including Midjourney-V5, Adobe Firefly and DALL-E 3. We present the comparison of results in Table 6. The numbers clearly demonstrate that the transfer learning strategies utilized in this paper surpass previously proposed deepfake detection methods. Additionally, it's noteworthy that our models are trained using only 200k *real/fake* images, compared to the studies we're comparing against, which utilize 720k images for training.

**Table 5: We present the results from our few-shot (32-shot) experiments, wherein we train CLIP using various transfer learning strategies on *real/fake* images from the ProGAN dataset. We then evaluate the trained models on images generated by GANs, diffusion models and commercial image generators.**

Method	Family of Generators			Average AP/Acc
	GAN AP/Acc	Diffusion AP/Acc	Comm. Tools AP/Acc	
Linear Probing	94.39 / 83.62	89.67 / 80.47	76.78 / <b>69.72</b>	86.95 / 77.94
Fine-tuning	97.09 / 85.23	90.14 / 77.18	71.35 / 65.90	86.19 / 76.11
Adapter	97.40 / 87.27	90.53 / 81.12	61.69 / 53.93	83.21 / 74.11
Prompt Tuning	<b>98.61</b> / <b>89.88</b>	<b>95.97</b> / <b>84.76</b>	<b>87.23</b> / 66.38	<b>93.94</b> / <b>80.34</b>

**Table 6: Robustness of transfer learning strategies across different families of generative models.**

Method	Family of Generators			Average AP/Acc
	GAN AP/Acc	Diffusion AP/Acc	Comm. Tools AP/Acc	
Wang et al. (CVPR'20)	92.32 / 78.23	74.29 / 57.15	61.57 / 52.43	76.06 / 62.61
Gragn. et al. (ICME'21)	98.88 / 92.83	92.86 / 64.53	72.58 / 56.53	88.10 / 71.30
Corvi et al. (ICASSP'23)	99.14 / 90.67	86.06 / 57.15	66.40 / 54.62	83.87 / 67.48
Ojha et al. (CVPR'23)	95.39 / 86.13	93.66 / 82.77	75.26 / 68.42	88.10 / 79.11
Ours (LP)	98.22 / 90.02	95.60 / 86.01	81.95 / 72.53	91.92 / 82.86
Ours (FT)	99.32 / 89.73	97.72 / <b>92.59</b>	<b>98.52</b> / <b>94.48</b>	<b>98.52</b> / <b>92.27</b>
Ours (Adapter)	99.63 / 94.13	96.83 / 85.70	70.29 / 55.17	88.92 / 78.33
Ours (Prompt T.)	<b>99.61</b> / <b>94.16</b>	<b>97.97</b> / 86.86	85.71 / 59.62	94.43 / 80.21

## 5 CONCLUSION

Our study examines the robustness of CLIP in detecting deepfake imagery across diverse data distributions. We explore four distinct transfer learning strategies, including Fine-tuning, Linear Probing, Prompt Tuning and training an Adapter Network, using a diverse training set of 200k images from the ProGAN dataset. Our experiments encompass evaluation on a comprehensive test set comprising 21 different image generators.

Through experiments, we illustrate that transfer learning strategies incorporating both the image and text components of CLIP consistently surpass the performance of simpler approaches like Linear Probing, which solely utilizes the visual aspect of CLIP. Our findings highlight Prompt Tuning's superiority over current baselines and SOTA methods, achieving significant margins of improvement while showcasing its efficacy despite minimal training parameters. Additionally, we conduct few-shot experiments, analyze robustness under post-processing operations such as JPEG compression and Gaussian blurring, and demonstrate the consistent performance of our CLIP-based detectors even with a smaller training set size of 20k images.

## ACKNOWLEDGMENTS

This research was supported by industry partners and the Research Council of Norway with funding to MediaFutures: Research Centre for Responsible Media Technology and Innovation, through the Centres for Research-based Innovation scheme, project number 309339; and (2) NORDIS 2, European Horizon 2020 grant number 101158604.

We extend our gratitude to the authors of [2, 8, 9, 14, 33, 46, 51] for sharing their codes, pre-trained models and collected datasets. We also acknowledge the use of Generative AI [35] for checking and correcting the grammar of this paper. However, it is important to note that we did not use the tool to generate totally new text, rather we use it to check/correct the grammar of our provided text.

## REFERENCES

- [1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems* 35 (2022), 23716–23736.
- [2] Quentin Bammey. 2023. Synthbuster: Towards detection of diffusion model generated images. *IEEE Open Journal of Signal Processing* (2023).
- [3] Andrew Brock, Jeff Donahue, and Karen Simonyan. 2018. Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096* (2018).
- [4] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. 2022. Efficient geometry-aware 3D generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16123–16133.
- [5] Liang Chen, Yong Zhang, Yibing Song, Lingqiao Liu, and Jue Wang. 2022. Self-supervised learning of adversarial example: Towards good generalizations for deepfake detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 18710–18719.
- [6] Yunjin Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. 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.
- [7] Umur Aybars Ciftci, Ilke Demir, and Lijun Yin. 2020. Fakecatcher: Detection of synthetic portrait videos using biological signals. *IEEE transactions on pattern analysis and machine intelligence* (2020).
- [8] Riccardo Corvi, Davide Cozzolino, Giada Zingarini, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. 2023. On the detection of synthetic images generated by diffusion models. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.
- [9] Davide Cozzolino, Giovanni Poggi, Riccardo Corvi, Matthias Nießner, and Luisa Verdoliva. 2023. Raising the Bar of AI-generated Image Detection with CLIP. *arXiv preprint arXiv:2312.00195* (2023).
- [10] Boris Dayma, Suraj Patil, Pedro Cuena, Khalid Saifullah, Tanishq Abraham, Phuc Le Khac, Luke Melas, and Ritobrata Ghosh. 2021. DALL-E Mini. <https://doi.org/10.5281/zenodo.5146400>
- [11] Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems* 34 (2021), 8780–8794.
- [12] Patrick Esser, Robin Rombach, and Björn Ommer. 2021. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 12873–12883.
- [13] Hongchang Gao, Jian Pei, and Heng Huang. 2019. Progan: Network embedding via proximity generative adversarial network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1308–1316.
- [14] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. 2023. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision* (2023), 1–15.
- [15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems* 27 (2014).
- [16] Diego Gragnaniello, Davide Cozzolino, Francesco Marra, Giovanni Poggi, and Luisa Verdoliva. 2021. Are GAN generated images easy to detect? A critical analysis of the state-of-the-art. In *2021 IEEE international conference on multimedia and expo (ICME)*. IEEE, 1–6.
- [17] Ishaa Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. *Advances in neural information processing systems* 30 (2017).
- [18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
- [19] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*. PMLR, 4904–4916.
- [20] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. 2022. Visual prompt tuning. In *European Conference on Computer Vision*. Springer, 709–727.
- [21] Tero Karras, Timo Aila, Samuli Laine, and Jaakkko Lehtinen. 2017. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196* (2017).
- [22] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakkko Lehtinen, and Timo Aila. 2020. Training generative adversarial networks with limited data. *Advances in neural information processing systems* 33 (2020), 12104–12114.
- [23] Tero Karras, Miika Aittala, Samuli Laine, Erik Häkkinen, Janne Hellsten, Jaakkko Lehtinen, and Timo Aila. 2021. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems* 34 (2021), 852–863.
- [24] Tero Karras, Samuli Laine, and Timo Aila. 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.
- [25] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakkko Lehtinen, and Timo Aila. 2020. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8110–8119.
- [26] Sohail Ahmed Khan and Duc-Tien Dang-Nguyen. 2023. Deepfake Detection: Analysing Model Generalisation Across Architectures, Datasets and Pre-Training Paradigms. *IEEE Access* (2023).
- [27] Konwoo Kim, Michael Laskin, Igor Mordatch, and Deepak Pathak. 2021. How to adapt your large-scale vision-and-language model. (2021).
- [28] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.
- [29] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *Comput. Surveys* 55, 9 (2023), 1–35.
- [30] Yisroel Mirsky and Wenke Lee. 2021. The creation and detection of deepfakes: A survey. *ACM Computing Surveys (CSUR)* 54, 1 (2021), 1–41.
- [31] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. 2021. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741* (2021).
- [32] Augustus Odena, Christopher Olah, and Jonathon Shlens. 2017. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*. PMLR, 2642–2651.
- [33] Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. 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.
- [34] OpenAI. 2021. guided-diffusion. <https://github.com/openai/guided-diffusion>.
- [35] OpenAI. 2021. Introducing ChatGPT. <https://openai.com/blog/chatgpt>. [Online; accessed 08-August-2023].
- [36] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. 2019. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2337–2346.
- [37] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. 2023. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952* (2023).
- [38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR, 8748–8763.
- [39] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* 1, 2 (2022), 3.
- [40] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 10684–10695.
- [41] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. 2019. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*. 1–11.
- [42] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems* 35 (2022), 36479–36494.
- [43] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training gans. *Advances in neural information processing systems* 29 (2016).
- [44] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114* (2021).
- [45] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. PMLR, 2256–2265.
- [46] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. 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.

- [47] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 8798–8807.
- [48] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. 2023. DIRE for Diffusion-Generated Image Detection. *arXiv preprint arXiv:2303.09295* (2023).
- [49] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. 2022. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7959–7971.
- [50] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. 2015. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365* (2015).
- [51] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Learning to prompt for vision-language models. *International Journal of Computer Vision* 130, 9 (2022), 2337–2348.
- [52] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.
- [53] Xiangyu Zhu, Hao Wang, Hongyan Fei, Zhen Lei, and Stan Z Li. 2021. Face forgery detection by 3d decomposition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2929–2939.