Universal deepfake detection across various image generators based on data from diffusion models

Karolina Łęcka[1] and Andrzej Rusiecki[2][0000−0003−2239−1076]

[1] Nokia, Wroclaw, Poland, karolina.lecka@nokia.com
[2] Faculty of Information and Communication Technology,
Wroclaw University of Science and Technology, Wroclaw, Poland,
andrzej.rusiecki@pwr.edu.pl

Abstract. Deepfake images, with artificially generated or partially altered fake content, appear more and more frequently in everyday life. In order to avoid misleading or misinforming potential recipients of such data, deepfake detection techniques are continuously developed and improved. Such approaches need to reliably detect images from various generation methods based on modern deep neural network architectures, such as generative adversarial networks (GAN) or diffusion models (DM). In this paper, we present an experimental analysis of the challenges associated with the detection of fake images generated from different sources. Inspired by previous research in this area, our experiments demonstrate the results of training classifiers using only the images generated by chosen diffusion models, as opposed to training detection models exclusively on images produced by GANs. We conducted the experiments on GenImage dataset using the state-of-the-art CLIP+ViT-L/14 backbone as feature extractor, combined with the recently proposed frequency masking approach. A comparative analysis revealed that a training set made up of diffusion model images can increase the average precision across a wide range of generators, resulting in a higher degree of model generalization. Moreover, training the detectors on DM data can result in relatively high accuracy in detecting also GAN-based deepfake images.

Keywords: Deepfake detection · Diffusion models · GANs · GenImage

1 Introduction

Artificially generating or altering existing images has become a popular and generally available process. Several easily accessible applications have emerged that leverage the available technology and generate any image based on a verbal description or a previously provided sketch, allowing many to create content of an entertaining or educational nature.

Unfortunately, it is often the case that the material generated is intended to manipulate, denigrate, ridicule, or misinform. Such use of technology is extremely problematic, as it involves showing people in untrue, often compromising situations or attributing to them words they never said. When using images of public figures, a carefully crafted deepfake can mislead and cause confusion and

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

misunderstanding. Such posts can spread on social media on a huge scale and unsuspecting users will then become part of the information war. In order to protect against manipulation and other harmful effects of the spread of artificial images, it is extremely important to develop methods to detect such fakes.

1.1 Deepfake generation

With the advancement of AI technology and the increase in computing power of available hardware, deepfake image generation is becoming more and more accessible and accurate in convincingly reproducing reality. Deep learning methods employed to deepfake generation are in general based on three types of network architectures: variational autoencoders (VAE), generative adversarial networks (GAN) and diffusion models (DM).

Autoencoders are amongst the first algorithms used to create deepfake images [10]. In such architectures the task of the encoders is to create a latent space of hidden variables, which contains detailed information about the elements of the image, such as the color and texture of the skin of the person depicted, the position of the head or the color of the hair. In VAEs, Gaussian distribution is used as the distribution of latent features in the generated space [21].

GANs are based on architectures consisting of a generator and a discriminator. These two modules compete with each other, resulting in an improvement in the quality of the generated images, as well as the ability to assign the correct class to the final image [16].

Probabilistic diffusion models have recently shown great results in the field of image generation. The process of image generation by DMs models is shown in Fig. 1. The first, forward stage (Fig. 1a) perturbs the input data by gradually adding noise, the second, backward stage (Fig. 1b) performs denoising in an iterative manner so as to optimize the parameters of the neural network. The last stage (Fig.1c), sampling, is responsible for using the optimized network to generate new data [3].

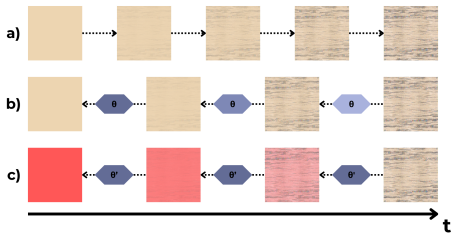

Fig. 1. Illustration of probabilistic DM image generation: a) forward processing - adding noise, b) backward processing - denoising, c) sampling and data generation.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

1.2 Deepfake detection

The most common approach to deepfake detection is the use of neural networks applied as feature extractors [13], or final neural classifiers [20, 19, 2, 11, 9, 22, 6]. A relatively new idea is to explore the frequency domain where significant artifacts have been observed - remnants of the use of generative models of various types. Exploration of second-order statistics has shown that these residuals are reflected in the spectral analysis of the images. This means that generative models are unable to fully reproduce the ratio of frequencies comprising a given image, which is an important clue for detection models [4].

In order to detect image manipulation, the authors of [20] propose convolutional neural networks, where in the signal amplification layer, a Laplace filter is used to amplify the differences between the images. The approach presented in [19] is based on Gaussian blur used for preprocessing to improve the statistical similarity at the pixel level between true and false images. The paper [2] describes the use of a convolutional layer tailored specifically for the task of deepfake image detection. It is supposed to autonomously learn what features indicate tampering with the image content, which would increase the generalization of the algorithm. In [11], the presented approach is based on the analysis of image depth maps and their anomalies, which may indicate the artificial origin of the data, while in [9] the main ideas is to use Siamese networks and training based on differences and similarities in the input images. The paper [22] describes an approach that is based on extracting features that indicate different origins of different parts of an image. By measuring the constancy of these features, it is possible to determine whether the image was modified.

One of the the latest approaches to deepfake detection [6] is based on the *masked signal modeling* method. It involves blocking part of input signals and trying to predict the content that should be in the masked areas. Different variants can be considered: masking individual pixels in the spatial domain, masking part of the image in the spatial domain, and masking in the frequency domain. The best results were obtained for masking in the frequency domain. The success of this method is related to the artifacts lef by generative models in the frequency domain [4]. In [13] it is noted that neural classifiers trained to discriminate between images can focus on some features only and omit others. It depends on the backbone network architecture and the dataset used for training: its size and diversity can significantly affect feature preference. It was also shown that partitioning the feature space obtained with pretrained encoders from the CLIP and ViT-L/14 architectures allows better separation between sets of real and artificial images than the use of the models pre-trained on ImageNet.

Most of the currently presented solutions assume training the network only on images generated by GAN models (including [19], [6], [2], [21]). Moreover, the datasets containing large quantities of images from DMs are relatively new, so the artifacts of GAN-generated images have already been well studied and described, while the current state of knowledge regarding the artifacts left by diffusion models is not particualry deep [18, 15].

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

2 Tested approach

Current research in the field of artificial image detection allows us to conclude that generative models, including diffusion models, leave artifacts or traces that can often be observed in the frequency domain [4]. Focusing more attention on features other than image content can have a positive impact on the generalization capabilities of the algorithm. The network architecture presented in [6] used frequency domain signal masking to enhance the generalization and training capabilities, hence a similar network model, adapted to use images derived from diffusion models, was used for this study.

2.1 Backbone network

The feature space of the tested model is formed by the CLIP (Contrastive Language-Image Pre-training) vision encoder. It was originally created to combine text with their visual counterparts to enable classification of objects into previously unknown classes (zero-shot learning). It is also distinguished from other models by the inclusion of publicly available internet data in its training set. Such a dataset contains far more categories than the popular ImageNet [14]. As previous studies have shown, combining the CLIP encoder with the ViT-L/14 vision transformer [7] provides a good separation of the features of the artificial and real images. Compared to other backbone networks used for image feature extraction, CLIP+ViT-L/14 also achieved the highest average precision for the various test sets [13].

2.2 Frequency Masking

In accordance with the conclusions described in [6], the experiment used masks to hide the 15% frequency components of the image. Masking was not limited to a specific frequency range, as hiding individual ranges yielded worse results than allowing the entire spectrum. Frequency masks were implemented to cover a given percentage of the selected frequency band in the frequency domain, after applying the Fourier transform to the input image. The specific frequencies that were being masked were chosen randomly. Finally, the input image was multiplied by the mask and the inverse Fourier transform was performed.

2.3 Testing environment

The research described in this paper was performed on the Bem 2 cluster, computing resources provided by the Wroclaw Networking and Supercomputing Center. Nodes offering Nvidia Tesla P100 GPU resources, CUDA cores and Intel Xeon Gold 6126 CPUs running on AlmaLinux 8.6 operating system were used.

The entire data preparation and handling process, model building, training, and testing were implemented in Python 3.8.16. The environment was created according to the recommendations presented in [6]. The most important libraries included PyTorch 1.13.1 [1], Pillow 9.4, torchvision 0.14.1, and scikit-learn 1.2.2. We used CUDA Toolkit version 11.7 with the torch.cuda pack.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

3 Data sets

A sufficiently large dataset was needed to train and validate the neural network classifiers. According to the assumptions, it had to consist only of images from diffusion models. Both of these conditions were met by the GenImage [23] collection [Fig. 2].

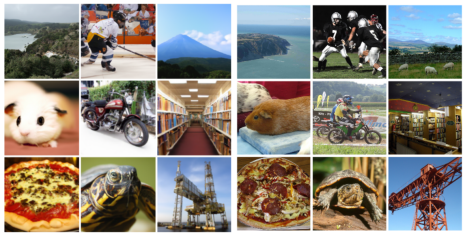

Fig. 2. Exemplary training images form the GenImage [23] dataset: artificial images (left), real images (right).

GenImage in its original form contains more than one million pairs of artificial and real images, a total of about 550 GB of data. It provides coverage of the 1,000 classes offered by ImageNet [5]. The images that are part of the collection come from 8 well-known and effective generators, namely: Midjourney, Stable Diffusion version 1.4 and 1.5, ADM, GLIDE, Wukong, VQDM, BigGAN. The latter is a GAN-type generator, so for the purposes of this study it was excluded from the training set. After removing images from the GAN model and corrupted files, the final training base had a total of 2 256 168 images, including 1 137 994 fake images and 1 119 174 real images. The validation base had a total of 88 014 images, including 44 007 artificial and real images each. The exact distribution of the number of images for each generator is presented in the Table 1. Files from the ImageNet collection are marked as "real". They are not shared between generators, which guarantees a higher degree of diversity in the database. Based on the class label of each real image, files described as "fake" were generated.

GenImage contains real images in JPEG format and artificial images in PNG format. This could potentially create conditions for the neural network to learn to classify samples based on artifacts resulting from compression. As the [8] shows, GenImage does indeed exhibit several predispositions for the results obtained using this collection to be biased. These are related, for example, to the previously mentioned compression or to the size of real and artificial images. This study, however, focuses on training the network on a set from a specific generator. In the case of JPEG compression, the application shows that trying to minimize this bias has little effect on the generalization capabilities of the detector. This may be related to the finding described in [4], which states that the bias is often already present in the generators themselves. Depending on the training set they use, they may attempt to reproduce JPEG compression

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

Table 1. Number of real and artificial images in the training and validation sets, divided into image generators.

Training set							
	Midjourney	**SD V1.4**	**SD V1.5**	**ADM**	**GLIDE**	**Wukong**	**VQDM**
real	161702	162001	153275	157454	162001	160740	162001
fake	161998	161998	166001	161995	162000	162001	162001
total	323700	323999	319276	319449	324001	322741	324002

Validation set							
	Midjourney	**SD V1.4**	**SD V1.5**	**ADM**	**GLIDE**	**Wukong**	**VQDM**
real	6001	6001	8001	6001	6001	6001	6001
fake	6001	6001	8001	6001	6001	6001	6001
total	12002	12002	16002	12002	12002	12002	12002

artifacts in their artificial images. This could be an important clue for image classification networks, as they can learn to distinguish between traces of true compression and attempts to reproduce them. In addition, it was shown that JPEG compression can almost completely obscure artifacts directly related to the artificial origin of the image, which would be observable in the autocorrelation function. Consistent with these findings, the entire training and validation set was not subjected to additional compression or selection.

The bias of the GenImage set related to the size of real and artificial data was also examined. The size of the real images in this set varies between 100 px and 1 550 px for both dimensions, with most having heights and widths falling within 300-400 px and 500-550 px, respectively. In [8] it was shown that an attempt to compensate for this bias can negatively affect the generalization of the final classifier. Hence, in our experiments, no additional selection of images based on their size was performed, and all images were cropped to a size of 224×224 px.

3.1 Test set

To test the quality of our approach, we decided to use the data sets that have already been used in testing deepfake detection methods.

Test set A. The first test set, hereinafter referred to as set A, is the image database proposed in [17] [Fig. 3]. It contained samples generated by the following models: ProGAN, BigGAN, StyleGAN, CycleGAN, GauGAN, StarGAN, DeepFake, SITD, SAN, CRN, IMLE.

The first three models represent unconditional GANs (the output images are created based on random noise). The next three instances are conditional GANs. They use an additional condition to generate images, which affects the content of the image. This can be, for example, text, another image, or the type of class to which the contents of the resulting file should belong. DeepFake contains artificial images derived from a model using an autoencoder, which have also undergone a number of additional transformations after generation. Seeing In The Dark (SITD) is a model created to improve the contrast and visibility

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

Fig. 3. Exemplary images from the first dataset used in [17]: artificial images (left), real images (right).

of elements in images that are too dark. It uses pairwise learning to extract features of correct exposure and apply them to a darkened image. Second Order Attention Network (SAN) is a network that allows to increase the resolution of images by using second-order statistics. It allows more accurate capture of the relationships of image structures. The last two generators, Cascaded Refinement Networks (CRN) and Implicit Maximum Likelihood Estimation (IMLE) are based on complex loss functions designed to map the way humans perceive images.

The test set A contained a total of 72 353 images, including 36181 real and 36 172 fake images - a total of about 19 GB of files. The exact distribution of the number of images for each generator is presented in the table below [Tab. 2]:

Table 2. Number of real and artificial images in the test set A, divided into image generators.

	ProGAN	CycleGAN	BigGAN	StyleGAN	GauGAN	StarGAN	DeepFake	SITD	SAN	CRN	IMLE
real	4000	1321	2000	5991	5000	1999	2707	180	219	6382	6382
fake	4000	1321	2000	5991	5000	1999	2698	180	219	6382	6382
total	8000	2642	4000	11982	10000	3998	5405	360	438	12764	12764

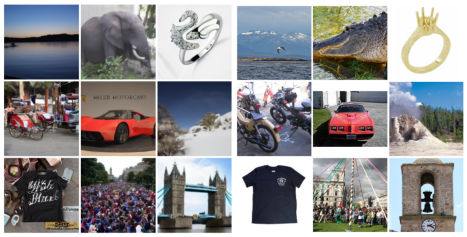

Fig. 4. Exemplary images from the test set B [13]: fake images (left), real images (right).

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

Test Set B. The second collection, refferd to as set B, was previously used in [6] and is an excerpt from the image database derived from the [13] [Fig. 4]. It focused on diffusion models (LDM, Glide, Guided Diffusion) and an autoregressive model (DALL-E-mini). Each category contained 1 000 artificial image files. The Guided Diffusion model was trained using ImageNet, so the real images were selected from that repository. The other models used real images from the LAION collection, which corresponded to the generated files. In total, the collection took up about 1 GB.

4 Experimental settings

Our experiments were performed on the datasets described in previous sections, with some additional preprocessing steps and hyperparameter tuning.

4.1 Data preprocessing

All the images were converted to the RGB color space, and then selected transformations were performed. These included applying frequency masks, and data augmentation techniques, such as resizing the image using bilinear interpolation, blurring using Gaussian noise, JPEG compression to a specified compression ratio, cropping to the selected size, flipping the image horizontally, and normalization. . The augmentation operations were independent of each other and applied to images with probability 0.1 for most of them and 0.5 for the horizontal flip. Finally, each image was cropped to 224×224 px, as described in the 3 section. The images of the training set were cropped at a random location, and those of the validation set were cropped, preserving the center part.

4.2 Model validation and testing

The first step before training the network was feature extraction from previously preprocessed images. It was performed using the CLIP+ViT-L/14 model, as described in section 2.1. Such feature extraction was the most time-consuming process: it took about 18.5 hours to extract features into 17 635 batches, containing features from 128 images each. The processing time was directly scaled with the amount of training data (more than 2.2 million images) and the batch size. Such feature spaces were created for the training and validation sets offline and then stored in appropriate files. After feature extraction, individual training and validation batches were transferred to the model.

In order to avoid overfitting, an early stopping approach was applied. We used a scheme where if the validation accuracy within a given number of epochs did not improve by at least 0.001 percentage points, the learning rate was reduced 10 times until a minimum value of $1e-6$ was reached. We used the $AdamW$ optimizer, which implements the independence of the regularization of the weights from the initial learning rate and has been shown to have a positive effect on the generalization of classifiers [12]. The *BCEWithLogitsLoss* (Binary Cross Entropy With Logits Loss) was chosen as the loss function.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

Model parameters Based on our preliminary tests and hyperparameter search, the following model configuration was chosen: frequency masking with the mask size 15 and masking all frequencies, batchsize 128, 500 epochs training with early stopping (no improvement in validation within 15 epochs), and initial learning rate: 0.0002. Exemplary training results were presented in Figure 5.

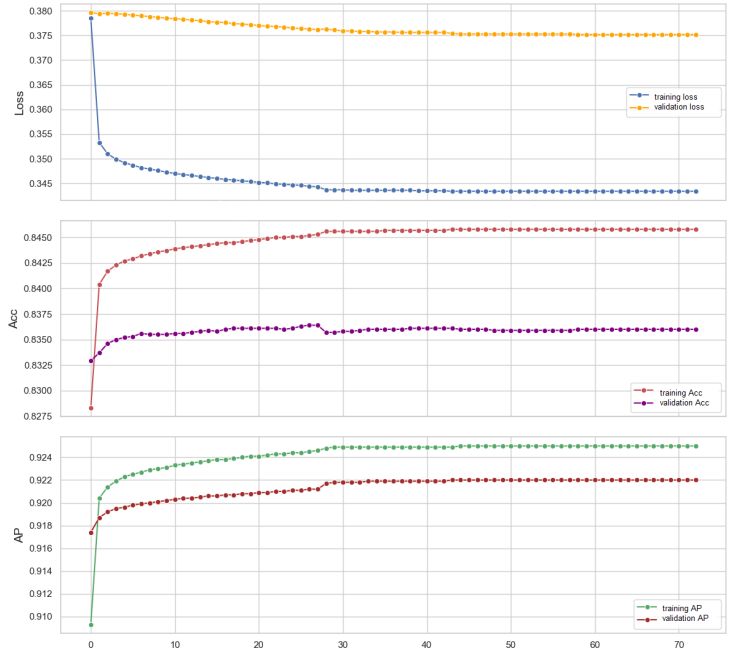

Fig. 5. Final model training progress ($batchsize = 128$, early stopping $patience = 15$).

4.3 Model testing

In the final model test, the data preprocessing was much simpler than for the model training and the choice of its configuration. It consisted only of cropping the center of the images to dimensions of 224×224 px and normalizing the tensors with values selected for the CLIP+ViT-L/14 network. The data prepared in this way was passed to the network model. This process, as in the case of training and validation, took into account the size of the minibatches - for the test set the batch size was chosen as 64. The effectivness of tested approaches was evaluated with basic metrics: accuracy (ACC), average precision (AP), and the area under the ROC curve (ROC AUC).

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

Table 3. Results of deepfake detectors tested on the test set A.

model	ProGAN	StyleGAN	BigGAN	CycleGAN	GauGAN	StarGAN	DeepFake	SITD	SAN	CRN	IMLE	average for all generators
batch_size = 128, patience = 15												
AP	0.987	0.908	0.966	0.945	0.954	0.976	0.731	0.809	0.908	0.810	0.957	**0.905**
ACC	0.878	0.690	0.895	0.843	0.752	0.920	0.667	0.636	0.801	0.632	0.851	**0.779**
ROC AUC	0.986	0.906	0.967	0.952	0.962	0.976	0.726	0.820	0.923	0.824	0.958	**0.909**

5 Experimental results

Tables 3 and 4 summarize the results of testing the deepfake detector on test sets A (generators unknown to the classifier) and B (data from these generators were included in the training set).

5.1 Detection on data generated by models unfamiliar to the classifier

The first of the test sets, set A, contained data from models completely unknown to the trained classifier, so it can be used to assess how well the deepfake detector can generalize over new data.

Analyzing the results in Table 3, it can be concluded that the model has achieved very good results for images derived from unconditional GANs. For the three models in this category - ProGAN, BigGAN and StyleGAN, the averaged precision and average ROC AUC does not fall below 95% in any case. An average accuracy of 82% was also achieved. Images generated by conditional GANs are also recognized very well. The averaged precision for CycleGAN, GauGAN and StarGAN does not fall below 94% in any case. An average precision of 81-82% and an average ROC AUC of 95% were also obtained.

A subset of images generated by the Deepfake model obtained the lowest AP and ROC AUC results. This part of the test set contained images of modified faces, while the training set did not focus on specific content. However, the abilities of the presented approach to recognize artificial patterns from this set exceed the random chance by 23 percentage points.

The results obtained for image enhancement models (SITD and SAN) are also very good. However, a difference between the accuracy of SITD and SAN can be noticed. The average precision and the area under the ROC curve are also quite different, again indicating a better match for the SAN generator. This result may be related to the specifics of SITD, where the model is concerned with improving the exposure of a photo. Correcting the brightness and contrast of an image most often does not involve modifying its content, which can result in completely different artifacts.

For images from generators intended to reproduce human perception, i.e. CRN and IMLE, the results obtained also confirm the good performance of the presented classifier. Here, too, significant differences can be observed between the metrics of the two generative models, which are also evident in the results presented in [6].

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

Table 4. Results of deepfake detectors tested on the test set B.

model	Guided	LDM_200	LDM_200_cfg	LDM_100	Glide_100_27	Glide_50_27	Glide_100_10	DALL-E	average for all generators
batch_size = 128, patience = 15									
AP	0.958	0.898	0.811	0.903	0.934	0.937	0.946	0.912	**0.912**
ACC	0.793	0.785	0.669	0.796	0.850	0.854	0.879	0.799	**0.803**
ROC AUC	0.963	0.908	0.818	0.910	0.945	0.947	0.954	0.922	**0.921**

5.2 Detection on data generated by models known to the classifier

During training, the network had access to images generated by various diffusion models, however, the images in the test set A were different from those in the training set. The data can therefore be considered as those to which the network had at least partial access previously, by training on data generated by similar models.

As can be seen in Table 4, very high AP, ACC and ROC AUC scores were obtained for the images created by the Guided Diffusion model. An interesting observation arises from the analysis of the classifier results for the LDM. Removing the restriction to generate images only from the previously described classes that were used in the process of training the generator, made correct classification much more difficult. This is most likely related to going beyond the classes defined by the ImageNet. No significant effect of the number of steps used to generate images on the classification results was observed.

Similar conclusions can be drawn when analyzing the Glide model. Changing the number of steps performed in the first stage does not significantly affect the results obtained. The number of iterations of the second stage of the Glide generator shows some correlation with the quality of detection: the accuracy of the tested solution increases as the number of steps in the second stage decreases. This leads to the conclusion that an improvement in the quality of the artificial image can make it more challenging to recognize.

For the DALL-E data, the results achieved show that the model's ability to detect deepfakes is very good. All three of the metrics tested are at a high level for each variant of the experiment.

5.3 Comparison with previous research

To compare our results with the state-of-the-art approaches, following metrics were used: average classification accuracy - Avg. acc (averaged over all test sets), and average precision - mAP (also averaged over all test sets). For the best hyperparamters (batch size = 128 and patience equal to 15) we obtained $AP = 0.908$ and $Acc = 0.789$.

The results were compared with the results described in [6] and [13], on which the concept of the present study was based (Tab. 5). The average precision for all image generators from the test sets improved after applying training on images from diffusion models by nearly 1.7%*pt*. As expected, an increase in classification precision was observed for almost all diffusion models. For Guided Diffusion it was as much as 8.57%*pt* improvement. The exception is the results

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

Table 5. Comparison of the detection results (mAP) with SOTA classifiers from [6] and [13].

	Tes set A										
Generator	ProGAN	StyleGAN	BigGAN	CycleGAN	GauGAN	StarGAN	DeepFake	SITD	SAN	CRN	IMLE
Ojha 2023 [13]	**1.000**	0.907	0.961	**0.992**	0.998	0.987	**0.773**	0.672	0.748	0.730	0.940
Doloriel + Ojha 2024 [6]	**1.000**	**0.927**	**0.969**	0.990	**0.998**	**0.989**	0.772	0.674	0.758	0.788	**0.959**
Our approach	0.987	0.908	0.966	0.945	0.954	0.976	0.731	**0.801**	**0.908**	**0.810**	0.957
	Test set B										
Generator:	Guided	LDM 200	LDM 200 CFG	LDM 100	Glide 100 27	Glide 50 27	Glide 100 10	DALL-E	Average over all image generators		
Ojha 2023 [13]	0.883	**0.955**	0.807	0.964	0.905	0.915	0.901	0.866	0.890		
Doloriel + Ojha 2024 [6]	0.818	**0.955**	0.809	**0.965**	0.907	0.917	0.902	0.867	0.893		
our approach	**0.958**	0.898	**0.811**	0.903	**0.934**	**0.937**	**0.946**	**0.912**	**0.908**		

for LDM without CFG - a decrease of about 6 percentage points was noted. Images generated by the autoregressive DALL-E model posed less of a challenge to the presented classifier - the average precision increased by more than 5%*pt*.

An obvious decrease in precision was observed for all GAN-type generators. Nevertheless, the average precision for the analyzed model did not fall below 90%, and the difference between the best result for a given GAN did not exceed 4.7%*pt*. The use of training images only from DMs negatively affected the classification precision for the DeepFake set. The biggest change was observed for the image enhancement models, SITD and SAN. The presented neural network achieved an average precision of almost 20%*pt* higher than the other solutions. These values suggest a link between the artifacts left by SITD and SAN and those of the diffusion models.

Statistical test. Performing statistical tests to compare the examined approaches, including results for all generators, seems to be useless, because increased performance for non-GAN images is associated with lower accuracy for GAN-based deepfakes. This is why we decided to compare the performance only for the non-GAN part of the whole test set. Assuming significance level $\alpha = 0.05$, we performed a non-parametric Friedman test obtaining $statistic = 7.412$, and $pvalue = 0.0246$, so we could reject the null hypothesis and state that the ranks measuring the performance of analyzed approaches differ. Hence, we could proceed with a post hoc Nemenyi test that revealed only a significant difference between our approach and [13] ($pvalue = 0.022$) while no difference was detected between [13] and [6] ($pvalue = 0.181$) or our approach and [6] ($pvalue = 0.638$).

6 Summary and conclusions

In this study we investigated the generalization ability of the SOTA deepfake image detectors when the training set consisted only of DM-generated images. We conducted the experiments on the GenImage dataset using the state-of-the-art CLIP+ViT-L/14 backbone as a feature extractor, combined with the recently proposed frequency masking approach. Such frequency masks were used to augment the training data in order to better expose the frequency domain of

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

the analyzed images. Combined with the feature space of the CLIP+ViT-L/14 model, the masking yielded very good results. A key conclusion that emerges from the performed experimental studies is that the artifacts left by diffusion models are relatively universal. There are probably some common patterns or features between them and artifacts left by GANs, autoencoders, or other types of image generators. A comparative analysis reveals that a training set made up of diffusion model images can increase the average precision across a wide range of generators, resulting in a higher degree of model generalization. Moreover, training the detectors on DM data can result in relatively high accuracy in detecting also GAN-based deepfake images.

References

1. Ansel, J., et al.: PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation. In: 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24). ACM (Apr 2024). https://doi.org/10.1145/3620665.3640366, https://pytorch.org/assets/pytorch2-2.pdf
2. Bayar, B., Stamm, M.C.: A deep learning approach to universal image manipulation detection using a new convolutional layer. In: Proceedings of the 4th ACM workshop on information hiding and multimedia security. pp. 5–10 (2016)
3. Chang, Z., Koulieris, G.A., Shum, H.P.: On the design fundamentals of diffusion models: A survey. arXiv preprint arXiv:2306.04542 (2023)
4. Corvi, R., Cozzolino, D., Poggi, G., Nagano, K., Verdoliva, L.: Intriguing properties of synthetic images: from generative adversarial networks to diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 973–982 (2023)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
6. Doloriel, C.T., Cheung, N.M.: Frequency masking for universal deepfake detection. arXiv preprint arXiv:2401.06506 (2024)
7. 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. arXiv preprint arXiv:2010.11929 (2020)
8. 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)
9. Hsu, C.C., Zhuang, Y.X., Lee, C.Y.: Deep fake image detection based on pairwise learning. Applied Sciences **10**(1), 370 (2020)
10. Katarya, R., Lal, A.: A study on combating emerging threat of deepfake weaponization (10 2020). https://doi.org/10.1109/I-SMAC49090.2020.9243588
11. Liang, B., Wang, Z., Huang, B., Zou, Q., Wang, Q., Liang, J.: Depth map guided triplet network for deepfake face detection. Neural Networks **159**, 34–42 (2023)
12. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8

13. Ojha, U., Li, Y., Lee, Y.J.: Towards universal fake image detectors that generalize across generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24480–24489 (2023)
14. 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: International conference on machine learning. pp. 8748–8763. PMLR (2021)
15. Ricker, J., Lukovnikov, D., Fischer, A.: AEROBLADE: Training-Free Detection of Latent Diffusion Images Using Autoencoder Reconstruction Error . In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9130–9140. IEEE Computer Society, Los Alamitos, CA, USA (Jun 2024). https://doi.org/10.1109/CVPR52733.2024.00872, https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.00872
16. Shen, T., Liu, R., Bai, J., Li, Z.: 'deep fakes' using generative adversarial networks (gan). Noiselab, University of California, San Diego (2018)
17. Wang, S.Y., Wang, O., Zhang, R., Owens, A., Efros, A.A.: Cnn-generated images are surprisingly easy to spot... for now. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8695–8704 (2020)
18. Wang, Z., Bao, J., Zhou, W., Wang, W., Hu, H., Chen, H., Li, H.: DIRE for Diffusion-Generated Image Detection . In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 22388–22398. IEEE Computer Society, Los Alamitos, CA, USA (Oct 2023). https://doi.org/10.1109/ICCV51070.2023.02051, https://doi.ieeecomputersociety.org/10.1109/ICCV51070.2023.02051
19. Xuan, X., Peng, B., Wang, W., Dong, J.: On the generalization of gan image forensics. In: Chinese conference on biometric recognition. pp. 134–141. Springer (2019)
20. Yang, P., Ni, R., Zhao, Y.: Recapture image forensics based on laplacian convolutional neural networks. In: Digital Forensics and Watermarking: 15th International Workshop, IWDW 2016, Beijing, China, September 17-19, 2016, Revised Selected Papers 15. pp. 119–128. Springer (2017)
21. Zendran, M., Rusiecki, A.: Swapping face images with generative neural networks for deepfake technology–experimental study. Procedia computer science **192**, 834–843 (2021)
22. Zhao, T., Xu, X., Xu, M., Ding, H., Xiong, Y., Xia, W.: Learning self-consistency for deepfake detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 15023–15033 (2021)
23. Zhu, M., Chen, H., Yan, Q., Huang, X., Lin, G., Li, W., Tu, Z., Hu, H., Hu, J., Wang, Y.: Genimage: A million-scale benchmark for detecting ai-generated image (2023)

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_8