Ukrainian Catholic University

Bachelor Thesis

AI-Generated Image Detection System for Mitigating Fake News and Misinformation

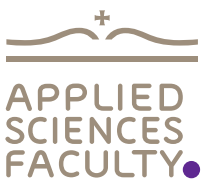
Author:
Bohdan Pelekh

Supervisor:
Anastasiia Mishchuk

A thesis submitted in fulfillment of the requirements for the degree of Bachelor of Science

in the

Department of Computer Sciences and Information Technologies
Faculty of Applied Sciences



Lviv 2025

"*I have not failed. I've just found 10,000 ways that won't work.*"

Thomas A. Edison

UKRAINIAN CATHOLIC UNIVERSITY

Faculty of Applied Sciences

Bachelor of Science

AI-Generated Image Detection System for Mitigating Fake News and Misinformation

by Bohdan Pelekh

Abstract

Despite numerous existing approaches, the problem of synthetic image detection remains unsolved—particularly in terms of generalisation to unseen generators and robustness to image distortions. At the same time, the use of synthetic images to create fakes and manipulations is becoming horrifying and dangerous. In this work, we investigate new methods for detecting synthetic images. We begin by utilizing pre-trained foundation models for visual representation, trained on real (DINOv2) and synthetic data (SynCLR). Then, we expand our approach with local patch texture analysis using a Local Binary Pattern (LBP) and a ResNet-18 model, analyzing the image from both global and local perspectives. We also created our custom evaluation dataset by collecting images from state-of-the-art generative models and focusing on topics that the most sensitive to manipulation and fakes. As a result, our best method surpasses the performance of the latest synthetic image detection techniques in terms of generalisation while maintaining a reasonable level of accuracy under common image distortions. This work shows the strengths and weaknesses of local features analysis, reveals the potential of visual representation models for synthetic image detection, and provides a valuable dataset of realistic synthetic images with an emphasis on their potential usage in fake news and misinformation.

The source code, weights are publicly available at GitHub: https://github.com/Bohdan213/thesis_synthetic_image_detection

Collected dataset is publicly available at Kaggle: https://www.kaggle.com/datasets/bohdan213/misinfo-dataset

Acknowledgements

I want to thank my supervisor, Anastasiia Mishchuk, for her support and valuable advice, without which such results would not have been achieved.

I extend my gratitude to the Faculty of Applied Sciences at UCU for providing the essential background for writing this work.

I express my deep gratitude to my parents for their immense contribution to my development and learning, for their constant support and love.

I want to thank my friends(485), without whom these four years would have been much tougher.

Finally, I am thankful to the Armed Forces of Ukraine and to all the Ukrainian people for their sacrifice and ongoing fight.

Contents

List of Figures

List of Tables

List of Abbreviations

CLIP	**C**ontrastive **L**anguage-**I**mage **P**re-training
CNN	**C**onvolutional **N**eural **N**etwork
DM	**D**iffusion **M**odel
GAN	**G**enerative **A**dversarial **N**etwork
JPEG	**J**oint **P**hotographic **E**xperts **G**roup
LBP	**L**ocal **B**inary **P**attern
LDM	**L**atent **D**iffusion **M**odel
LLM	**L**arge **L**anguage **M**odel
MLP	**M**ulti-**L**ayer **P**erceptron
VAE	**V**ariational **A**uto**E**ncoder
ViT	**V**ision **T**ransformer

Dedicated to my family and my hometown – Berdyansk

Chapter 1

Introduction

1.1 Motivation

What is typical for each technology, artificial intelligence brings us big opportunities which we could not imagine before. With the increased popularity of AI generative models that provide qualitative and realistic images and videos, people can easily convert their thoughts and ideas into art and distribute them across the world. However, everything that gives big opportunity - also brings significant responsibility.

After the release of the latest state-of-the-art image generation models, their usage has grown exponentially. We can observe synthetic content in social networks, media, and street banners. Last studies have shown that humans struggle significantly to distinguish real photos from AI-generated ones, with a misclassification rate of 38.7%. [22]

With each new model and improvements it's getting much harder to identify synthetic images from real ones. This leads to several significant issues that are raising up these days, such as:

- Fake images strengthen the spread of misinformation, fake news, and disinformation campaigns on social media [33].
- AI-generated images can pose a threat to scientific integrity
- The constant and imperceptible mixing of real and AI-generated images may hinder future research and training, emphasizing the need for effective filtering mechanisms to distinguish between them.

The problem of detecting realistic synthetic images also lies in the fact that generative models used for spreading fake news may be publicly unknown and closed to collecting samples for training. Moreover, real-world images may be affected by common distortions like JPEG compression or Gaussian blur, which could also totally mislead some detection approaches.

In this thesis, we aim to propose and explore a set of methods and identify ones that can effectively handle unseen generators and have a reasonable level of resilience to image distortions. Additionally, we aim to evaluate our approach using state-of-the-art image generation models available today.

1.2 Contributions

This work explores the advantages and limitations of using different foundation models for visual representations in the task of synthetic image detection. We propose an approach for extracting local texture features and combine it with a global-level view, resulting in a powerful two-branch model that operates on both global and local levels.

In more detail, the main contributions of this thesis are:

1. Collecting a dataset of synthetic images generated by the latest state-of-the-art models, with an emphasis on their potential use in fake news and misinformation.
2. Research of use of the foundational model for visual representations trained exclusively on synthetic data and a comparison with model pre-trained on real-image datasets.
3. A proposed method for extracting local texture features and applying it to the problem of synthetic image detection.
4. Two-branch solution that operates on both global and local levels.

1.3 Structure Of The Thesis

Thesis structured in the next way:

- **Chapter 2: Related Work**

  This chapter provides an overview of the field of synthetic image detection, describing the main research directions and their most influential works.

- **Chapter 3: Theoretical Background**

  This chapter presents the theoretical background necessary to understand the proposed approaches.

- **Chapter 4: Proposed Solution**

  This chapter provides a detailed description of our approach and the hypotheses behind it.

- **Chapter 5: Dataset**

  Here, we introduce our self-collected evaluation dataset and describe open-source dataset used for training and evaluation.

- **Chapter 6: Experiments**

  This chapter contains information about the experiments setup, their results, comparisons with related work, and a discussion of the findings.

- **Chapter 7: Conclusion**

  Finally, we conclude our work and discuss potential directions for future improvements.

Chapter 2

Related Work

2.1 Image Generation

Image generation has made significant progress in the last few years. The first modern and revolutionary approach to generate high quality and believable images is Generative Adversarial Network (GAN) [10], in which main idea lies in the simultaneous training of 2 neural networks - a generator and a discriminator. While the generator's primary goal is to create a realistic image, the discriminator aims to determine whether the image is real or was produced by the generator. These models train in adversarial mode, essentially the Min-Max game with the unique solution, where the generator learns the entire training distribution and the discriminator achieves a success rate of $\frac{1}{2}$. In practice, it is hard to achieve this due to issues like mode collapse, unstable training dynamics, the difficulty of balancing the two networks, and finally, the discriminator model winning the competition.

Another image generation approach that revolutionised the field through the simplicity of its training is the denoising diffusion models [13, 34, 37]. The principle is built on adding noise and further step-by-step recovering to generate a high-quality output. A single loss function enables better stability and allows the model to learn from a large amount of training data more efficiently than GANs. Almost all modern commercial and open-source image generation models are based on diffusion. They also utilise other techniques such as variational autoencoders (VAEs) [18] and guidance mechanisms [25].

2.2 Synthetic Image Detection

The development and proliferation of high-quality synthetic images pushed the research of various detection methods to identify generated content. A key challenge is ensuring generalisation to unseen patterns and robustness to JPEG compression and blur augmentations. Here, we discuss the main directions that have proven their efficiency and have attracted the greatest research efforts.

2.2.1 Spatial inconsistencies and artifacts detection

CNNSpot [42] was one of the early attempts to detect synthetic images. The authors focus on utilizing a simple Convolutional Neural Network (CNN) model to capture visible artifacts that deteriorate visual quality and frequency patterns that persist in various generation models. The model was trained exclusively on ProGAN [16] in a binary classification setup to detect real and fake examples, demonstrating strong generalisation capabilities across different GAN-based models.

Instead of analysing pixel-level or frequency-based artifacts, Chuangchuang Tan et. al [38] focus on gradient-based inconsistencies in GAN images. The pretrained transformation model converts images into gradients, filtering out content and retaining only discriminative pixels. Then, this gradients are used as a feature representation for a classifier that predicts whether the image is real or fake.

Another approach relies on a multi-view image completion representations followed by classification [21]. The authors train several encoder-decoder networks on real images to reconstruct missing information from different incomplete views (masked regions, grayscale images, edge-based representations) and learn the distribution of real images. After that, a classifier determines whether an image is real or fake based on the discrepancies detected in the reconstruction of each view.

However, these methods become useless in front of the new Diffusion Models (DM) in the context of generalisation to unseen models.

2.2.2 Diffusion reconstruction

Reliable detection and generalisation of images created with diffusion models require investigating approaches beyond simple spatial-based methods.

To address this generalisation challenge, Zhendong Wang et al. [43] propose a method based on image inversion and reconstruction using a different diffusion model, followed by a comparison between the original and reconstructed images.

The core idea behind DIRE [43] is the assumption that synthetic images generated by the diffusion process can be reconstructed more accurately by the diffusion model than real images.

After reconstruction, authors compute the difference between the original and reconstructed images to quantify the reconstruction error, which further serves as a discriminative feature for the classifier.

With the emergence of latent diffusion models, Jonas Ricker et al. [32] propose a training-free method that utilises LDMs for reconstruction, followed by threshold-based classification of the reconstruction error.

George Cazenavette et al. [2] also use the Latent Diffusion Model (LDM) but train a classification CNN using a concatenation of the input image, noise vector, and decoded image.

2.2.3 Local patches investigation

In addition to global-level features, researchers are also exploring local features obtained from smaller patches. For instance, Jiaxuan Chen et al. [3] focus on the noise pattern of a single patch with minimal texture diversity.

The core idea lies in the assumption that generative models often prioritise creating patches with detailed textures to enhance the realism of images while overlooking the subtle noise introduced by the camera capture, especially in simple patches.

To improve robustness against image degradation, the patch goes through the Enhancement Module and is then classified after applying a high-pass filter.

The approach proposed by Yan Ju et al. [15] employs a two-branch model where a Patch Selection Module extracts informative local patches, and an Attention-based Feature Fusion Module integrates spatial features from the entire image and local features via multi-head attention. This enhances generalisation to unseen generative models, however, the result is still far from ideal.

Zhong et al. [50] introduce a detector that emphasizes texture-based fingerprints by exploiting the contrast between rich and poor texture regions while breaking the

global semantic information. The authors use randomly cropped patches to divide the image into two parts: patches with high and low texture diversity. Then, they apply a set of high-pass filters proposed in SRM [9] to extract noise patterns, which are then processed by a lightweight convolutional network and used for classification.

2.2.4 Foundation models

While the previous approach tries to suppress the semantic information of the images, another lines of work attempt to leverage the advantages of vector representations from pretrained foundation models.

Ojha et al. [27] use the CLIP [31] feature space to distinguish between real and fake samples. The authors experimented with the k-nearest neighbors method and also implemented a simple linear separator added to the visual representation model. Their result shows that using a universal feature space not explicitly trained for real-vs-fake classification provides a good baseline for detecting fake images across diverse generation techniques.

Incorporating previous techniques, Shilin Yan et al. [46] propose a two-branch detector with both semantic and local patch-based methods. They combine semantic embeddings extracted from a CLIP-based encoder with low-level statistical features derived from rich and poor texture patches. Their approach achieves good generalisation to unseen models and better robustness to image augmentations.

Chapter 3

Theoretical Background

In this chapter, we provide an overview of the key concepts and methods that form the foundation of our approach. We begin by introducing DINOv2 [29], a powerful self-supervised vision model designed to extract high-quality, general-purpose visual features from large-scale image data. We then present SynCLR [39], a novel framework that shifts away from traditional real-world datasets by leveraging synthetic data for visual representation learning. We finish with a description of the Local Binary Pattern (LBP) that plays an important role in our solution.

3.1 DINOv2

DINOv2 [29] is a self-supervised learning approach for training vision model to produce high-quality visual representations. The core idea lies in creating a general-purpose foundation model that can be used across various image-related tasks without additional fine-tuning.

DinoV2 based on the 1B parameters ViT-G [8] model that later was distilled into smaller models (ViT-S, ViT-B, ViT-L) while retaining robust feature extraction capabilities. These models learn discriminative features at both the global image level and local patch level, making them well-suited for fine-grained analysis of textures, inconsistencies, and artifacts that are often present in AI-generated images.

Compared to weakly-supervised models like CLIP [31], OpenCLIP [5] and SWAG [36], which use text-image pairs, DINOv2 learns purely from images. For that, authors use the LVD-142M [29] dataset with 142 million diverse images, carefully curated to enhance training quality. Despite the absence of text supervision, DINOv2 matches or surpasses these weakly supervised models in several benchmarks.

3.2 SynCLR

Visual representation learning methods depend on large real-world datasets, which are difficult to curate and label at scale. To address this issue, Yonglong Tian et al. introduced SynCLR [39], a framework that learns visual representations using only synthetic images and synthetic captions. The whole training pipeline can be described in a few steps:

- Generate synthetic captions with Llama-2 7B [40] Large Language Model (LLM).
- Using synthetic captions and Stable Diffusion V1.5 [34] generate synthetic images
- Train visual representation model using contrastive learning [17] and masked image modeling [51].

This approach achieves strong performance and competes favorably with CLIP and DINOv2 and other state-of-the-art general-purpose visual representation learners.

3.3 Local binary pattern

Local Binary Pattern (LBP) [26] is a texture descriptor which is widely used for various image processing tasks. His main idea lies in comparing intensity of neighboring pixels with that of the central pixel. For each neighbor pixel we assign a binary value based on the fact whether his intensity is greater or smaller to the center pixel.

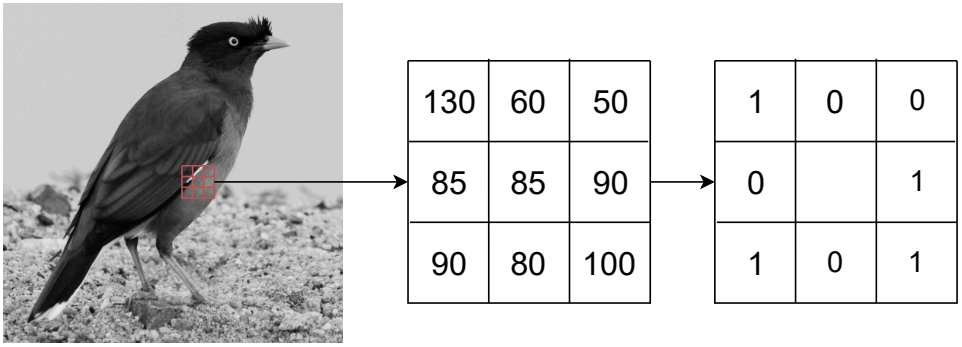
Figure 3.1 shows the algorithm of LBP computation. Given a grayscale image I, for each pixel at position (x_c, y_c), the LBP code is computed as:

$$\mathrm{LBP}_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(i_p - i_c) \cdot 2^p \tag{3.1}$$

Where:

- $i_c = I(x_c, y_c)$ is the intensity of the center pixel, and i_p is the intensity of the p-th neighboring pixel
- P is the number of neighbors
- R is the size of neighbors sliding window
- $s(x)$ is the thresholding function:

$$s(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases}$$



$$\mathrm{LBP}_{8,3}(x_c, y_c) = 1 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot 2^3 + 1 \cdot 2^4 + 0 \cdot 2^5 + 1 \cdot 2^6 + 0 \cdot 2^7 = 89$$

FIGURE 3.1: Local binary patterns computation.

Once the LBP code is computed for each pixel, we construct a new image where each pixel is replaced by its corresponding LBP value as shown in the Figure 3.2. Such representation effectively highlight local patterns such as edges, spots, and flat areas. Further analysis of such representations may reveal patterns and features that might not be visible in the original image.

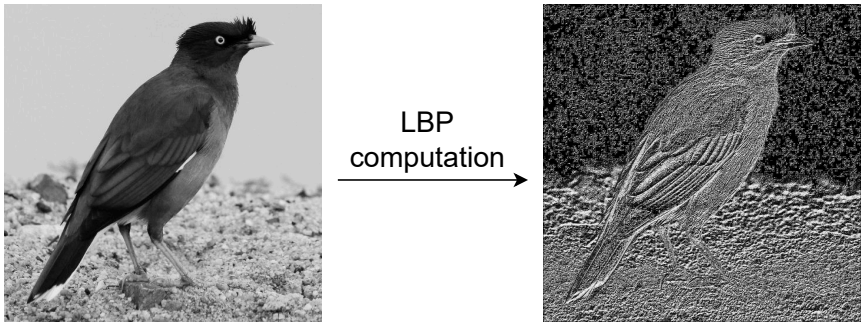


FIGURE 3.2: Local binary encoding for image.

Chapter 4

Proposed Solution

4.1 Problem Formulation

Based on the existing synthetic image detection approaches and evaluation benchmarks, we define the following objectives, which represent the main challenges in this task and which we aim to address:

- **Generalisation to Unseen Generators**: Ensuring that the proposed detection method preserves its effectiveness across unknown image-generation architectures that were not present during training.
- **Robustness to common distortions**: Method should retain a reasonable level of accuracy under JPEG compression and Gaussian blur distortions.
- **Balanced Dataset with State-of-the-Art Generative Models**: Construct new balanced datasets that include images generated by the latest state-of-the-art generative models and focus on topics most sensitive to manipulation in media and news.

4.2 Global level view using foundation model

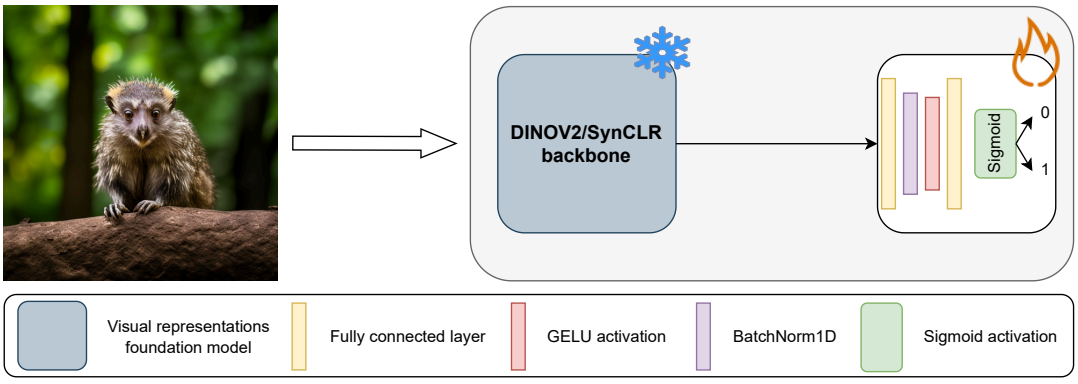


FIGURE 4.1: Overview of a fake/real classifier based on an MLP head placed on top of a backbone pretrained foundation model with frozen parameters.

To begin, we focus on the power of vector spaces provided by foundation models for visual representations, which are pre-trained on large-scale datasets. For our initial solution, we employ two foundation models described in Chapter 3 that share the same ViT-L/14 architecture but are trained on radically different data and using distinct training methods.

On top of each model, we place a simple Multi-Layer Perceptron (MLP) head for binary classification. The overall structure is shown in Figure 4.1.

4.2.1 Foundation model pretrained on real data

While related works use CLIP-based foundation models, we start with the DINOv2 [29] model, developed to effectively capture visual representations from images. We believe this model can detect spatial and frequency-based artifacts of synthetic images and provide well-separable vector representations that broadly cover the distribution of real images and distinguish unseen generators.

4.2.2 Foundation model pretrained on synthetic data

As described in Section 3.2, SynCLR [39] is a powerful foundation model pre-trained on the large number of synthetic images collected with Stable Diffusion V1.5 [34]. As a result, the model is totally unfamiliar with the distribution of real images, while it can identify and encode artifacts specific to synthetic data. We want to examine how well SynCLR can identify synthetic images and notice all the samples that do not belong to the cohort of real images.

To ensure fairness, we want all our solutions learned on images from only one generative model architecture throughout the training cycle. Therefore, we use samples from Stable Diffusion V1.4, which shares the same architecture with Stable Diffusion V1.5.

4.3 Global and local levels integration

Solutions based only on a global-level view may not be effective enough with unseen generators. Considering this, we complement our approach with a method that also works with a local-level view.

4.3.1 LBP for poor texture patch

Image generation models tend to focus on regions with rich textures, trying to recreate realistic shapes and edges as accurately as possible. In contrast, they pay significantly less attention to simpler regions that have smaller impact on the loss function.

To calculate a patch's texture diversity and corresponding poor texture regions, we adopt the approach proposed by Zhong et al. [50]. To evaluate the diversity of texture within an $M \times M$ patch, we compute the sum of absolute differences between neighbouring pixels along four directions: horizontal, vertical, diagonal, and counter-diagonal. The texture diversity score l_{div} is defined as:

$$\begin{aligned} l_{div} = &\sum_{i=1}^{M}\sum_{j=1}^{M-1}|x_{i,j} - x_{i,j+1}| + \sum_{i=1}^{M-1}\sum_{j=1}^{M}|x_{i,j} - x_{i+1,j}| \\ &+ \sum_{i=1}^{M-1}\sum_{j=1}^{M-1}|x_{i,j} - x_{i+1,j+1}| + \sum_{i=1}^{M-1}\sum_{j=1}^{M-1}|x_{i+1,j} - x_{i,j+1}| \end{aligned} \tag{4.1}$$

Where $x_{i,j}$ is the pixel within the patch. To get the patch with the poorest texture, we select one with the lowest l_{div}.

As described in Section 2.2.3, related works use SRM [9] filters to capture frequency patterns and noise residuals that are often ignored by generative models.

We offer another approach to representing and highlighting textural features — the Local Binary Pattern (LBP), the underlying principles described in Section 3.3. In combination with poor texture patches and threshold function:

$$s(x) = \begin{cases} 1, & \text{if } x < 0 \\ 0, & \text{otherwise} \end{cases}$$

sliding window of size 3 and 8 neighbours, we can easily compare such representations for real and synthetic images.

FIGURE 4.2: In the first column, we show the original image. The second column displays the simplest patch of size 32×32, and the third column presents its corresponding LBP representation visualised with the viridis colormap. The top three rows contain synthetic images generated by Stable Diffusion V1.4, while the bottom rows shows real images from ImageNet.

Figure 4.2 shows real and fake examples of LBP computation on the patch with the poorest textures. Real-world images exhibit more complex structures, non-repetitive patterns and natural noise fluctuations, while synthetic image textures tend to have more balanced distributions.

4.3.2 Two-branch solution with common MLP head

We propose adding a LBP representation of poor texture patches as second branch to visual representation model that works on global-level view.

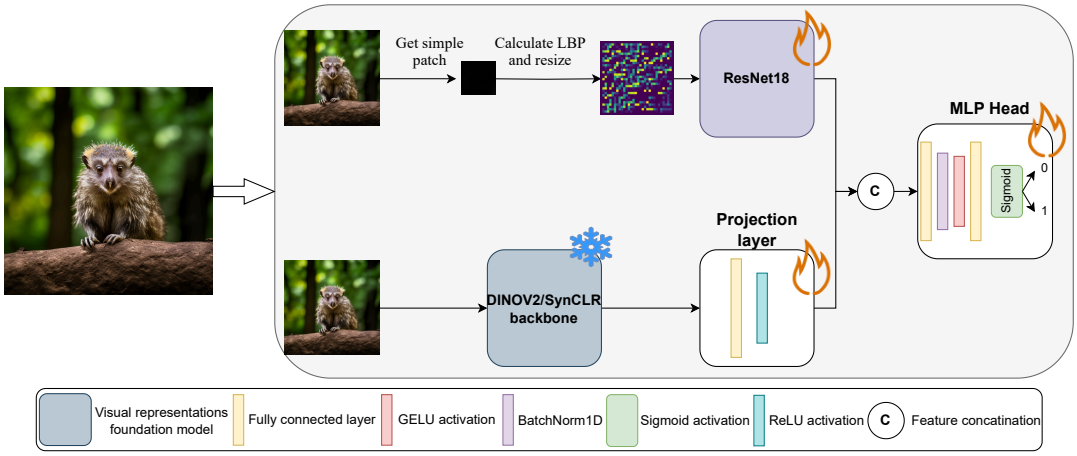


FIGURE 4.3: Overview of a fake/real classifier as a two-branch architecture with a common MLP head. Backbone pretrained foundation model has frozen parameters, while the other components are trainable for binary classification.

To extract the patch with the poorest texture, we randomly crop the image into N patches:

$$N = \left(\frac{\text{image\_width}}{M}\right) \times \left(\frac{\text{image\_height}}{M}\right),$$

where M is the patch size.

Then, we calculate the texture diversity for all patches and select the one with the lowest value.

The next step is to compute the patch's LBP representation and resize it to 256×256 using nearest-neighbor interpolation. To extract vectorised features, we utilise a simple ResNet-18 [12] model with no pre-trained weights.

For the global-level branch, we freeze the visual representation model's parameters and add a linear projection layer to align feature dimensions and facilitate effective fusion with local texture features.

Chapter 5

Datasets

5.1 Main Dataset

The GenImage [53] dataset is a large-scale benchmark for AI-generated image detection, comprising 1.33 million real images from ImageNet [6] and 1.35 million fake images generated using ImageNet class-based textual captions. GenImage covers a broad range of image categories and employs state-of-the-art generative models as of 2023, including GANs (e.g., BigGAN [1]) and diffusion models (e.g., Stable Diffusion V1.4 [34], Stable Diffusion V1.5 [34], Midjourney[23], ADM [7], Glide [24], Wukong [44], VQDM [11]).

The authors use ImageNet's distinct image classes to generate relevant semantically similar images. Each generative model is represented by approximately the same number of images per class — 162 for training and 6 for testing — except for Stable Diffusion V1.5, which is represented by 166 training images and 8 testing images per class.

The dataset contains good-quality, realistic images and is widely used for training and evaluation of various synthetic image detection methods. This makes it an excellent benchmark for comparing the performance of the proposed hypothesis against existing approaches.

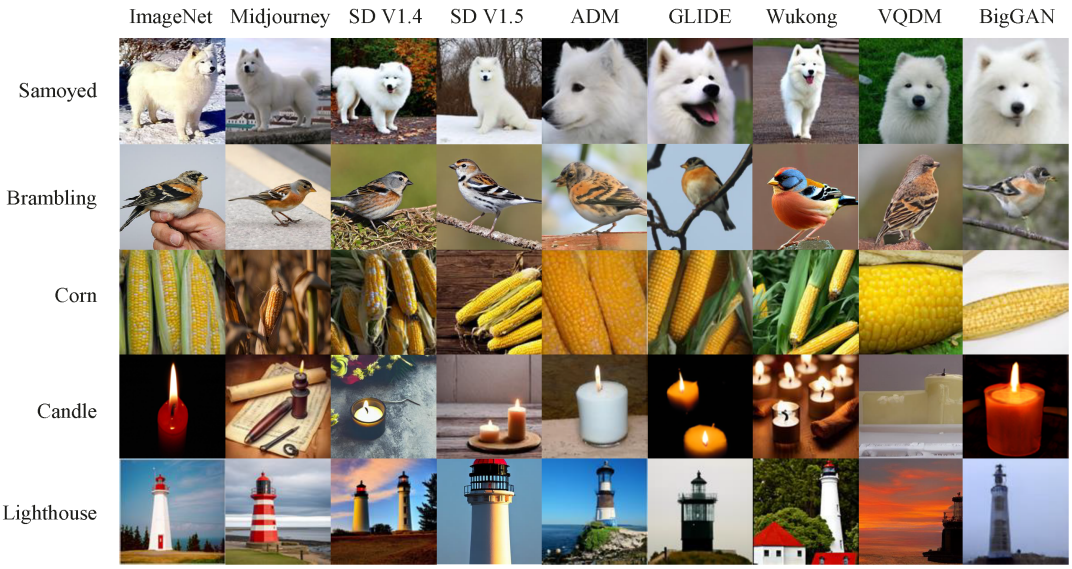


FIGURE 5.1: Visualization of images on GenImage dataset. Figure was taken from GenImage[53] original paper

We use the Stable Diffusion V1.4 training set for all our proposed methods to be consistent with related works. The training set consists of 162K real and 162K

synthetic images. For evaluation, we use each model's test set provided in the dataset, which contains 6K real and 6K synthetic images.

We also use a subset from the MidjourneyV5 training set for our additional experiment with foundation models. The training subset consists of 30K real and 30K synthetic images.

5.2 Custom dataset

5.2.1 Dataset objectives

Some state-of-the-art image generation solutions from 2023–2025 are not represented in any open-source datasets. Moreover, the images available in existing datasets do not fully align with the issue of misinformation and fake news and do not cover real-case fakes scenarios and sensitive topics. To fill this gap, we develop and introduce our self-collected evaluation dataset and test our hypotheses on this images as well.

5.2.2 Dataset collection

We selected 10 topics that are most sensitive to misinformation and fake news.

- Political and Government-Related Scenes
- Science and Technology-Related Scenes
- Military and Law Enforcement Scenes
- Natural Disasters and Emergency Situations
- Social and Economic Scenes
- Accidents
- Environmental and Ecological Scenes
- Sports and Entertainment Scenes
- Futuristic and Sci-Fi Scenes
- Wildlife and Nature Scenes

For each of the 10 topics, OpenAI's GPT [28] LLM generated 10 text prompts to serve as input for image generation, with 100 classes in total.

We employ four generative models for image generation: Imagen3 [14], SD-XL 1.0 [30], FLUX.1-dev [19], PixArt-α XL MS 2 [4]. Each model can generate high-quality realistic images that are almost indistinguishable to the human eye. Images from the Imagen3 were collected using the Google API, while the others were obtained via local inference. For each model, 1K 1024×1024 images were generated — 10 per class. All prompts were supplemented with words that enhance the realism of the generated images, along with negative prompts, with the goal of minimizing the possibility of producing painted or illustrative-style outputs.

To balance dataset we add real samples from the COCO[20] 2017 validation set that consist of 5000 images. All images with resolution lower than 400×400 were discarded. From the remaining images, we randomly selected 4K, to match the number of generated images in our dataset.

FIGURE 5.2: Visualization of synthetic images from four state-of-the-art e models — FLUX1-dev [19], Imagen3 [14], SDXL1.0 [30], and PixArt XL 2 MS [4] — on our custom dataset. Each row corresponds to a different prompt designed to reflect misinformation-prone scenarios.

As a result, we obtain a dataset, where each generative model is represented by 1K synthetic and 1K real images. Examples of generated images along with their corresponding text prompts can be found in Figure 5.2.

Chapter 6

Experiments and Results

6.1 Training setup

We train the global-level one-branch model proposed in Section 4.2 using both DINOv2 and SynCLR foundation models as the backbone. We adopt Adam optimiser with a learning rate of 1×10^{-4} for MLP head. The models are trained with a batch size of 64 for a single epoch, and the learning rate is reduced by half after half of the epoch.

The same combinations are used for the global-and-local two-branch solution described in Section 4.3. In this case, we also adopt the same foundation models and use a ResNet-18 [12] with randomly initialised weights — adapted for single-channel input — for local feature extraction. We adapt Adam optimiser with a learning rate of 1×10^{-4} for ResNet-18 and MLP head, and 5×10^{-5} for the projection layer following the foundation model. The training setup remains the same, with a batch size of 64 and one epoch of training. The learning rates are also reduced by half after half of the epoch.

For data augmentations we employ JPEG compression ($QF \sim \text{Uniform}(90, 100)$) and Gaussian blur ($\sigma \sim \text{Uniform}(0.1, 1)$) with 10% probability during all training sessions.

6.2 GenImage dataset evaluation

Method	Midjourney	SD v1.4	SD v1.5	ADM	GLIDE	Wukong	VQDM	BigGAN	Mean
ResNet-50 [12]	54.90	**99.90**	99.70	53.50	61.90	98.20	56.60	52.00	72.09
DeiT-S [41]	55.60	**99.90**	99.80	49.80	58.10	98.90	56.90	53.50	71.56
Swin-T [48]	62.10	**99.90**	99.80	49.80	67.60	99.10	62.30	57.60	74.78
CNNSpot [42]	52.80	96.30	95.90	50.10	39.80	78.60	53.40	46.80	64.21
Spec [45]	52.00	99.40	99.20	49.70	49.80	94.80	55.60	49.80	68.79
F3Net [47]	50.10	**99.90**	**99.90**	49.90	50.00	**99.90**	49.90	49.90	68.69
GramNet [49]	54.20	99.20	99.10	50.30	54.60	98.90	50.80	51.70	69.85
DIRE [43]	60.20	**99.90**	99.80	50.90	55.00	99.20	56.00	50.20	70.66
UnivFD [27]	73.20	84.20	84.00	55.20	76.90	75.60	56.90	80.30	73.29
GenDet [52]	**89.60**	96.10	96.10	58.00	78.40	92.80	66.50	75.00	81.56
PatchCraft [50]	79.00	89.50	89.30	77.30	78.40	89.30	83.70	72.40	82.30
ESSP [3]	82.60	99.20	99.30	78.90	88.90	98.60	96.0	73.90	90.60
AIDE [46]	79.38	99.74	99.76	78.54	91.82	98.65	80.26	66.89	86.88
DinoV2	78.95	97.22	96.71	59.60	79.22	92.49	72.19	76.18	81.60
DinoV2+LBP 1patch	83.43	96.78	96.68	83.98	92.51	95.82	93.44	81.96	90.58
DinoV2+LBP 2patches	84.76	97.56	97.39	**84.55**	**93.21**	97.03	**94.42**	**83.23**	**92.14**
SynCLR	72.98	99.6	99.54	50.64	52.82	94.88	49.92	49.86	71.5
SynCLR+LBP 1patch	80.07	99.78	99.67	54.35	68.25	98.12	62.21	50.38	76.60
SynCLR+LBP 2 patches	80.57	99.87	96.71	53.07	67.52	98.37	59.88	50.33	76.16

TABLE 6.1: Performance comparison of detectors across GenImage [53] synthetic generators at a 0.5 threshold. All detectors were trained using Stable Diffusion V1.4 [34]. The best result and the second-best result are marked in **bold** and underline. All evaluation sets have the same support for real and synthetic examples — 6K each.

The evaluation results of our methods, along with other synthetic image detectors, are presented in Table 6.1. The proposed methods are marked in bold. We utilise a two-branch model that captures both local and global features and evaluates it in two configurations: (1) regular inference using a single simple patch and (2) test-time two patches ensembling, where features from the last and second-to-last patches with poor texture are averaged before classification.

SynCLR demonstrates significantly weaker generalisation capabilities compared to DINOv2 across both proposed architectures at the decision threshold 0.5. At the same time, it achieves better performance when evaluated on its own training generator's architecture and similar architectures. This behaviour probably suggests that the model overfitted to the Stable Diffusion architecture, as it was pre-trained for visual representation on images generated by the same model.

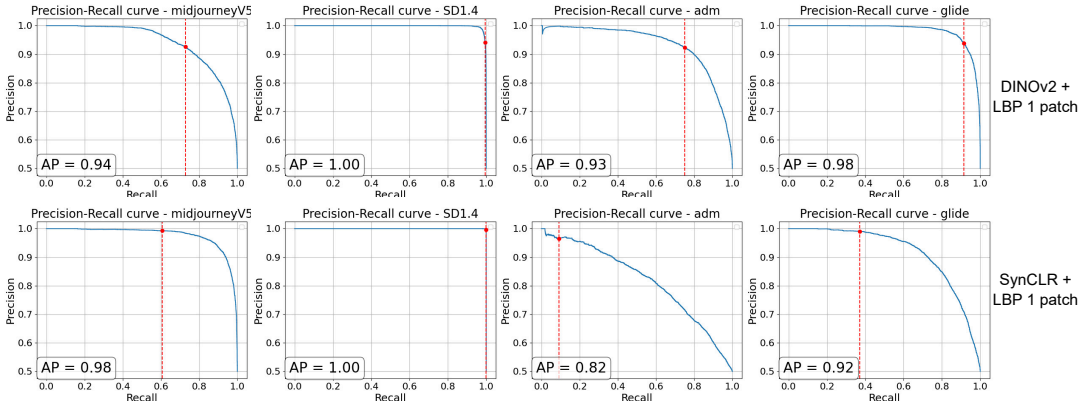


FIGURE 6.1: Precision-recall curves for MidjourneyV5 [23], Stable Diffusion V1.4 [34], ADM [7] and GLIDE [24] generation models from GenImage dataset. Red vertical line corresponds to 0.5 decision threshold.

Figure 6.1 presents precision-recall curves and average precision metrics for two two-branch architectures with different backbone models. Grounded on these plots, we observe that the DINOv2-based method exhibits, on average, relatively higher curves and better precision and achieves better precision and recall levels across different thresholds. Even with a standard threshold of 0.5, the model archives a good balance between first and second-type errors.

At the same time, the method with SynCLR as the backbone shows, on average, a lower precision-recall curve on unseen generators. We also see that model has low confidence for positive class, indicating that it requires further calibration of the decision threshold to achieve better generalization accuracy.

Additionally, results show that architecture with DINOv2 as a backbone achieves a higher Average Precision (AP) score than SynCLR-based.

Summarising, we observe a significant improvement in accuracy when comparing our purely global approach with the combined global and local levels views. Incorporating a poor-texture patch represented using LBP gives **+8.98% acc** and **+10.54% acc** with test-time two-patch ensembling, when using the DINOv2-based architecture as the backbone. Similarly, the SynCLR-based architecture achieves **+5.1% acc** and **+4.66% acc** improvements, respectively. With confidence, proposed two-branch architecture has excellent generalisation capabilities to unseen generators.

6.3 Custom dataset evaluation

Method	FLUX.1-dev	Imagen3	SDXL-1.0	PixArt-α XL2	Mean
SSP[3]	98.60	99.0	94.60	94.45	96.54
AIDE[46]	87.50	84.80	96.66	87.55	89.13
DINOv2	70.40	60.35	73.9	82.55	71.80
DINOv2+LBP 1 patch	<u>86.75</u>	<u>90.85</u>	93.75	85.75	<u>89.28</u>
DINOv2+LBP 2 patches	**87.95**	**93.05**	<u>95.55</u>	**88.35**	**91.26**
SynCLR	64.45	54.80	72.15	74.95	66.50
SynCLR+LBP 1 patch	73.65	79.75	95.00	86.95	83.80
SynCLR+LBP 2 patches	72.95	78.95	**95.65**	<u>87.95</u>	83.88

TABLE 6.2: Performance comparison of detectors across synthetic generators in our self-collected dataset at a 0.5 threshold. All detectors were trained using Stable Diffusion V1.4 [34]. The best result and the second-best result among our proposed methods are marked in **bold** and <u>underline</u>, respectively. All evaluation sets have the same support for real and synthetic examples — 1K each.

We also use our novel self-collected dataset to evaluate two related state-of-the-art synthetic image detection methods, which also operate at local and both global and local levels, along with our proposed methods.

We ought to highlight that the SSP method proposed by Jiaxuan Chen et al. [3] differs from the ESSP variant presented in Table 6.1. ESSP includes an enhanced network designed to improve robustness to image distortions, but the authors have not yet provided code implementation. As a result, the original SSP demonstrates very limited robustness to various types of distortions.

Our two-branch approach with DINOv2 as backbone still outperform other proposed solutions and AIDE [46], another method that operates at both global and local levels views. SSP is the undisputed leader, but its robustness against JPEG compression and Gaussian blur distortions without the enhanced network is on very low level.

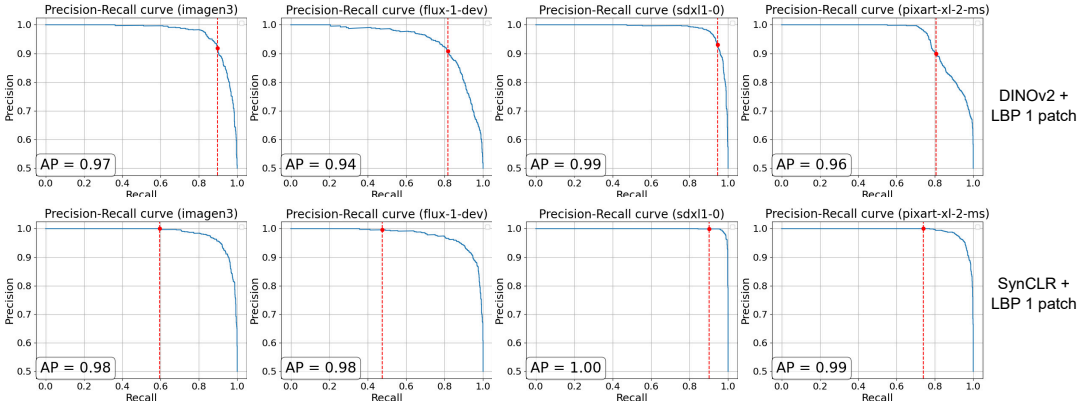


FIGURE 6.2: Precision-recall curves for FLUX1-dev [19], Imagen3 [14], SDXL-1.0 [30] and PixArt-α XL 2 [4] generation models from our self collected dataset. Red vertical line corresponds to 0.5 decision threshold.

Figure 6.2 presents precision-recall curves and average precision metrics for two two-branch architectures with different backbone models on our self-collected dataset.

Here, we see slightly different results than with GenImage dataset. The SynCLR-based method exhibits higher and more convex curves, especially when compared to DINOv2. We also observe higher average precision scores across all generators for a method based on a synthetic foundation model. These results suggest that SynCLR has greater potential for detecting more recent and high-realistic synthetic images than DINOv2. However, it still has low confidence for synthetic images and requires additional threshold calibration and adjustment to find optimal one.

Based on these findings, we see that modern image generation methods continue to differ from real images and can be effectively detected without the need for explicit training on their samples.

Additionally, Figure C.1 shows examples where the two-branch architecture with DINOv2 as the backbone failed.

6.4 Robustness to distortions

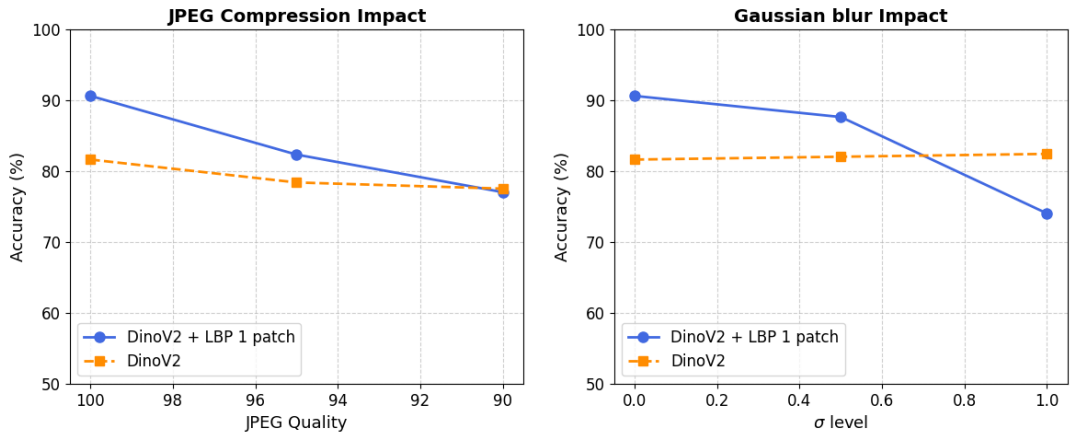


FIGURE 6.3: Robustness of two our proposed architectures with DINOv2 as backbone to JPEG compression and blur. Models were tested across all synthetic image generators in the GenImage [53] dataset. Plots represent the average accuracy across these generators.

Real-life images can be affected by various types of perturbations. We check our methods on two of the most common ones — JPEG compression and Gaussian blur. As presented in Figure 6.3 we evaluate one and two-branch methods with DINOv2 as backbone.

Both methods suffer from JPEG compression. The two-branch model that works with textures at the local level loses a relatively higher percentage of generalisation than the global-level solution. Regarding Gaussian blur, we see good robustness with only the foundation model as the backbone, and accuracy decreases with a method that operates at global and local levels. However, we still observe a reasonable level of accuracy, which, even with distortions, is greater than most other synthetic image detection methods.

The comparatively higher accuracy loss in the global and local views method is likely due to disruptions in pixel distributions and the elimination of discriminative artifacts, which are present in real and synthetic images. So, we can assume that although local features improve the level of generalisation to unseen generators, they have stronger vulnerabilities to image distortions.

Chapter 7

Conclusions

7.1 Results Summary

In this thesis, we propose new approaches for synthetic image detection problem, that analyse image either at the global level or at both global and local levels. Our best solution achieves the highest accuracy performance in the context of generalisation to unseen generators across various state-of-the-art detection methods.

We also compared two different foundation models for visual representation, used as backbones in the proposed methods. Our results demonstrate that both DINOv2 and SynCLR exhibit strong potential for detecting synthetic images, with SynCLR being our preferred choice for future investigations.

We propose the Local Binary Pattern for effective texture representations at the local level. We show that local level investigations have advantages in detecting unseen image generation architectures, but are vulnerable to image distortions.

Furthermore, we collect and use in the evaluation a new dataset with samples from the state-of-the-art generative models released in 2023–2025, most of them are not included in any existing benchmark at the time of writing this thesis. This dataset focuses on the topics most commonly associated with fake news and is designed to support the evaluation of solutions to mitigate such types of manipulation.

7.2 Contribution

This work makes a significant contribution to the field of synthetic image detection:

- We collected a new benchmark dataset that does not just contain synthetic images on topics lacking semantic sense but includes realistic images that can be used to spread misinformation.
- We propose a novel architecture that operates at both global and local levels, achieving superior performance compared to related works. We also present a comprehensive set of architectural variations and show great potential for ablation studies.

7.3 Future work and possible improvements

There is a huge room for improvement in the proposed solutions and the collected dataset.

Although the synthetic images in our dataset are focused on fake news topics, the same cannot be said for the real samples used to balance the dataset. A possible improvement would be extracting real images that are semantically similar to the

prompts used for generating fake images using a large image-caption dataset such as LAION [35].

Another important aspect that requires further attention is improving robustness to distortions. Images in the wild are often degraded during transmission or user interaction, so detection methods should maintain even better resilience. One possible solution is to use only global-level approaches, which are less sensitive to such perturbations. For our methods, it could also be beneficial to apply stronger data augmentations during training or to incorporate enhancement networks such as the one proposed by Jiaxuan Chen et al. [3].

Appendix A

Ablation study: rich texture patch

A.1 Rich texture patches

In this ablation study, we research how much the performance of our solution changes when using a rich texture patch instead of a poor one. We use the same two-branch architecture with DINOv2 foundation model as the backbone and the same training setup.

A.1.1 GenImage dataset evaluation

Method	Midjourney	SD v1.4	SD v1.5	ADM	GLIDE	Wukong	VQDM	BigGAN	Mean
DinoV2	78.95	97.22	96.71	59.60	79.22	92.49	72.19	76.18	81.60
DinoV2+LBP rich texture patch	78.11	97.28	97.26	63.54	79.60	95.72	76.67	79.35	83.4
DinoV2+LBP poor texture patch	83.43	96.78	96.68	83.98	92.51	95.82	93.44	81.96	90.58

TABLE A.1: Performance comparison of detectors based on DINOv2 and have different local levels, including rich texture patch, across GenImage [53] synthetic generators at a 0.5 threshold. All detectors were trained using Stable Diffusion V1.4 [34]. All evaluation sets have the same support for real and synthetic examples — 6K each.

Results presented in Table A.1 confirm the assumption that poor texture patches are more informative in the context of synthetic image detection problem. However, they also exhibit a small improvement compared to global-level-only architecture.

A.1.2 Custom dataset evaluation

Method	FLUX.1-dev	Imagen3	SDXL-1.0	PixArt-α XL2	Mean
DINOv2	70.40	60.35	73.9	82.55	71.80
DINOv2+LBP rich texture patch	65.15	69.70	81.75	78.00	73.65
DINOv2+LBP poor texture patch	86.75	90.85	93.75	85.75	89.28

TABLE A.2: Performance comparison of detectors based on DINOv2 and have different local levels, including rich texture patch, across synthetic generators in our self-collected dataset at a 0.5 threshold. All detectors were trained using Stable Diffusion V1.4 [34]. All evaluation sets have the same support for real and synthetic examples — 1K each.

Evaluation results on state-of-the-art image generation models presented in Table A.2 confirm the results presented in Section A.1.1.

Appendix B

Ablation study: further foundation models exploring

Results presented in Section 6 show that the proposed two-branch architectures with SynCLR backbone have great potential to detect more recent and high-realistic synthetic images. In this ablation study, we perform additional research of the one-branch architecture proposed in Section 4.2 using either DINOv2 or SynCLR as the backbone. Both models are trained on images from the highest-quality synthetic images generator in the GenImage dataset — MidjourneyV5 [23] — and evaluated on our self-collected dataset.

As described in Section 4.2.2, in this case, the solution based on the SynCLR model will know about two different generators — Stable Diffusion V1.5, on which the foundation model was pre-trained and MidjourneyV5. This comparison is slightly unfair to DINOv2, which will see samples from only one generator - MidjourneyV5. That is why it only goes after the main work.

B.1 Training setup

For training, use a subset from the MidjourneyV5 training set, 30K real and 30K synthetic images. We adopt the same Adam optimiser with a learning rate of 1×10^{-4} for MLP head. The models are trained with a batch size of 64 for two epochs, and the learning rate is reduced by half after each epoch.

For data augmentations we employ JPEG compression ($QF \sim \text{Uniform}(90, 100)$) and Gaussian blur ($\sigma \sim \text{Uniform}(0.1, 1)$) with 10% probability during all training sessions.

B.2 Custom dataset evaluation

Method	FLUX.1-dev	Imagen3	SDXL-1.0	PixArt-α XL2	Mean
DINOv2	87.70	72.40	82.55	92.10	83.69
SynCLR	98.10	77.15	93.95	98.50	91.93

TABLE B.1: Performance comparison of detectors across synthetic generators in our self-collected dataset at a 0.5 threshold. All detectors were trained using MidjourneyV5 [23]. All evaluation sets have the same support for real and synthetic examples — 1K each.

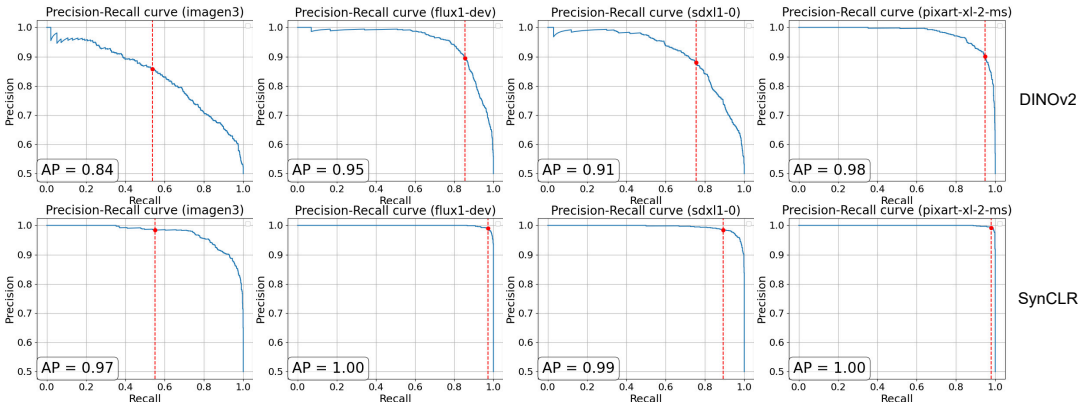


FIGURE B.1: Precision-recall curves for FLUX1-dev [19], Imagen3 [14], SDXL-1.0 [30] and PixArt-α XL 2 [4] generation models from our self collected dataset. Red vertical line corresponds to 0.5 decision threshold. Both detection methods were trained on MidjourneyV5 [23].

Accuracy results presented in Table B.1 and precision-recall curves with AP scores shown in Figure B.1 confirm assumption that bare SynCLR outperforms DINOv2 in the detection of synthetic images from latest generation models. However, it is worth recalling that in this experiment, SynCLR saw two different generation architectures, even in an implicit way, while DINOv2 — only one.

Appendix C

Failed examples

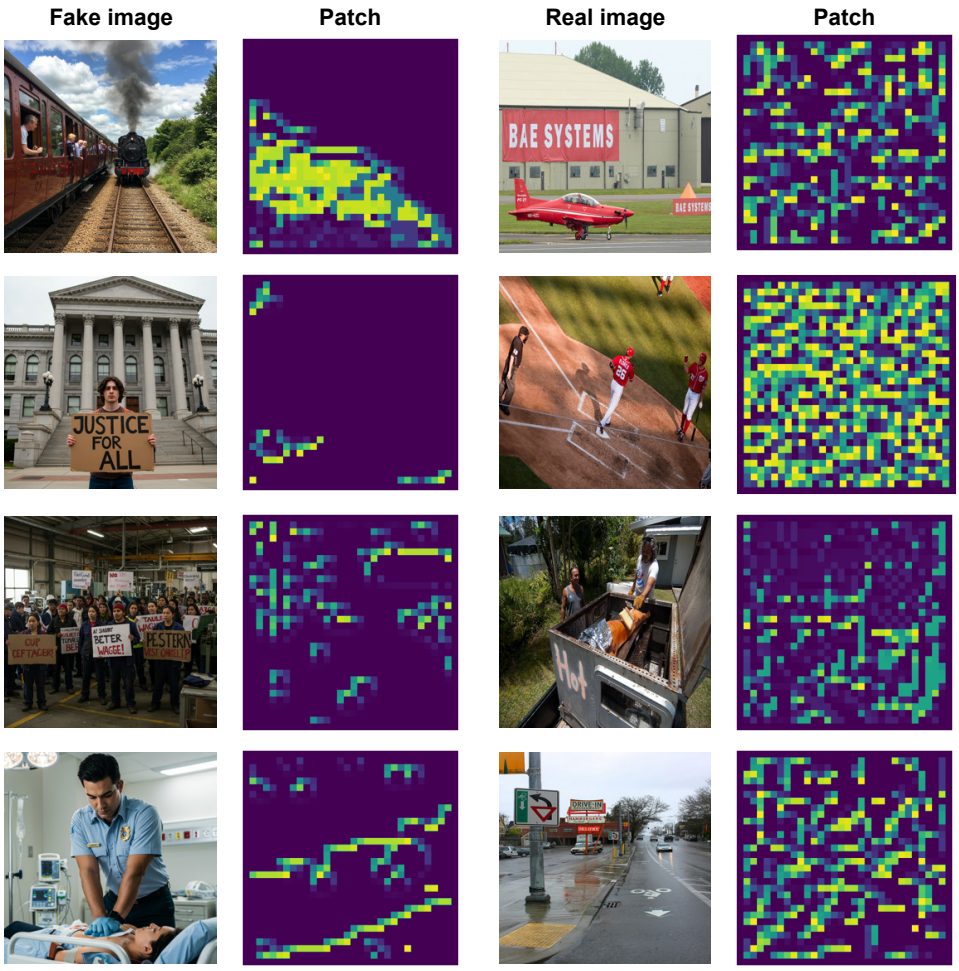


FIGURE C.1: Examples of failed classification of **DINOv2+LBP 1patch** on synthetic images from Imagen3 [14] and real images from COCO [20].

Bibliography

[1] Andrew Brock et al. *Large Scale GAN Training for High Fidelity Natural Image Synthesis*. 2019. arXiv: 1809.11096 [cs.LG].

[2] George Cazenavette et al. "FakeInversion: Learning to Detect Images from Unseen Text-to-Image Models by Inverting Stable Diffusion". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2024, pp. 10759–10769. DOI: 10.1109/CVPR52733.2024.01023.

[3] Jiaxuan Chen et al. *A Single Simple Patch is All You Need for AI-generated Image Detection*. 2024. arXiv: 2402.01123 [cs.CV].

[4] Junsong Chen et al. *PixArt-α: Fast Training of Diffusion Transformer for Photorealistic Text-to-Image Synthesis*. 2023. arXiv: 2310.00426 [cs.CV].

[5] Mehdi Cherti et al. "Reproducible scaling laws for contrastive language-image learning". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 2818–2829. DOI: 10.1109/CVPR52729.2023.00276.

[6] Jia Deng et al. "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[7] Prafulla Dhariwal et al. "Diffusion Models Beat GANs on Image Synthesis". In: *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2021, pp. 8780–8794. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/49ad23d1ec9fa4bd8d77d02681df5cfa-Paper.pdf.

[8] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. arXiv: 2010.11929 [cs.CV].

[9] Jessica Fridrich et al. "Rich Models for Steganalysis of Digital Images". In: *IEEE Transactions on Information Forensics and Security* (2012), pp. 868–882. DOI: 10.1109/TIFS.2012.2190402.

[10] Ian J. Goodfellow et al. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML].

[11] Shuyang Gu et al. "Vector Quantized Diffusion Model for Text-to-Image Synthesis". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 10686–10696. DOI: 10.1109/CVPR52688.2022.01043.

[12] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[13] Jonathan Ho et al. "Denoising Diffusion Probabilistic Models". In: *Advances in Neural Information Processing Systems*. 2020, pp. 6840–6851. URL: https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

[14] Google Imagen Team. *Imagen 3*. 2024. arXiv: 2408.07009 [cs.CV].

[15] Yan Ju et al. "Fusing Global and Local Features for Generalized AI-Synthesized Image Detection". In: *2022 IEEE International Conference on Image Processing (ICIP)*. 2022, pp. 3465–3469. DOI: 10.1109/ICIP46576.2022.9897820.

[16] Tero Karras et al. *Progressive Growing of GANs for Improved Quality, Stability, and Variation*. 2018. arXiv: 1710.10196 [cs.NE].

[17] Prannay Khosla et al. "Supervised Contrastive Learning". In: *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2020, pp. 18661–18673. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf.

[18] Diederik P. Kingma et al. *Auto-Encoding Variational Bayes*. 2022. arXiv: 1312.6114 [stat.ML].

[19] Black Forest Labs. *Announcing FLUX.1: A New Era in Text-to-Image Generation*. 2024. URL: https://bfl.ai/announcements/24-08-01-bfl.

[20] Tsung-Yi Lin et al. "Microsoft COCO: Common Objects in Context". In: *European Conference on Computer Vision (ECCV)*. Springer, 2014, pp. 740–755. DOI: 10.1007/978-3-319-10602-1_48.

[21] Chi Liu et al. "Towards Robust Gan-Generated Image Detection: A Multi-View Completion Representation". In: *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 2023. DOI: 10.24963/ijcai.2023/52.

[22] Zeyu Lu et al. "Seeing is not always believing: Benchmarking Human and Model Perception of AI-Generated Images". In: *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2023, pp. 25435–25447. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/505df5ea30f630661074145149274af0-Paper-Datasets_and_Benchmarks.pdf.

[23] *MidJourney*. https://www.midjourney.com/home. 2022.

[24] Alex Nichol et al. *GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models*. 2022. arXiv: 2112.10741 [cs.CV].

[25] Alexander Nichol et al. "Improved Denoising Diffusion Probabilistic Models". In: *Proceedings of the 38th International Conference on Machine Learning*. 2021, pp. 8162–8171. URL: https://proceedings.mlr.press/v139/nichol21a.html.

[26] Timo Ojala et al. "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2002), pp. 971–987. DOI: 10.1109/TPAMI.2002.1017623.

[27] Utkarsh Ojha et al. "Towards Universal Fake Image Detectors that Generalize Across Generative Models". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 24480–24489. DOI: 10.1109/CVPR52729.2023.02345.

[28] OpenAI. *ChatGPT*. Large language model. 2024. URL: https://chat.openai.com/chat.

[29] Maxime Oquab et al. *DINOv2: Learning Robust Visual Features without Supervision*. 2024. arXiv: 2304.07193 [cs.CV].

[30] Dustin Podell et al. *SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis*. 2023. arXiv: 2307.01952 [cs.CV].

[31] Alec Radford et al. "Learning Transferable Visual Models From Natural Language Supervision". In: *Proceedings of the 38th International Conference on Machine Learning*. 2021, pp. 8748–8763. URL: https://proceedings.mlr.press/v139/radford21a.html.

[32] Jonas Ricker et al. "AEROBLADE: Training-Free Detection of Latent Diffusion Images Using Autoencoder Reconstruction Error". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2024, pp. 9130–9140. DOI: 10.1109/CVPR52733.2024.00872.

[33] Jonas Ricker et al. "AI-Generated Faces in the Real World: A Large-Scale Case Study of Twitter Profile Images". In: *Proceedings of the 27th International Symposium on Research in Attacks, Intrusions and Defenses*. Association for Computing Machinery, 2024, 513–530. DOI: 10.1145/3678890.3678922.

[34] Robin Rombach et al. "High-Resolution Image Synthesis with Latent Diffusion Models". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 10674–10685. DOI: 10.1109/CVPR52688.2022.01042.

[35] Christoph Schuhmann et al. *LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs*. 2021. arXiv: 2111.02114 [cs.CV].

[36] Mannat Singh et al. "Revisiting Weakly Supervised Pre-Training of Visual Perception Models". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 794–804. DOI: 10.1109/CVPR52688.2022.00088.

[37] Jiaming Song et al. *Denoising Diffusion Implicit Models*. 2021. arXiv: 2010.02502 [cs.LG].

[38] Chuangchuang Tan et al. "Learning on Gradients: Generalized Artifacts Representation for GAN-Generated Images Detection". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 12105–12114. DOI: 10.1109/CVPR52729.2023.01165.

[39] Yonglong Tian et al. "Learning Vision from Models Rivals Learning Vision from Data". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2024, pp. 15887–15898. URL: https://cvpr.thecvf.com/virtual/2024/poster/29614.

[40] Hugo Touvron et al. *Llama 2: Open Foundation and Fine-Tuned Chat Models*. 2023. arXiv: 2307.09288 [cs.CL].

[41] Hugo Touvron et al. "Training data-efficient image transformers & distillation through attention". In: *Proceedings of the 38th International Conference on Machine Learning*. 2021, pp. 10347–10357. URL: https://proceedings.mlr.press/v139/touvron21a.html.

[42] Sheng-Yu Wang et al. "CNN-generated images are surprisingly easy to spot... for now". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020. DOI: 10.1109/CVPR42600.2020.00872.

[43] Zhendong Wang and Hong Chen others Hezhen Hu. "DIRE for Diffusion-Generated Image Detection". In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 22388–22398. DOI: 10.1109/ICCV51070.2023.02051.

[44] *Wukong*. https://xihe.mindspore.cn/modelzoo/wukong/introduce. 2022.

[45] Zhang Xu et al. "Detecting and Simulating Artifacts in GAN Fake Images". In: *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*. 2019, pp. 1–6. DOI: 10.1109/WIFS47025.2019.9035107.

[46] Shilin Yan et al. *A Sanity Check for AI-generated Image Detection*. 2025. arXiv: 2406.19435 [cs.CV].

[47] Qian Yuyang et al. "Thinking in Frequency: Face Forgery Detection by Mining Frequency-Aware Clues". In: *Computer Vision – ECCV 2020*. Springer International Publishing, 2020, pp. 86–103. DOI: 10.1007/978-3-030-58610-2_6.

[48] Liu Ze et al. "Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows". In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 9992–10002. DOI: 10.1109/ICCV48922.2021.00986.

[49] Liu Zhengzhe et al. "Global Texture Enhancement for Fake Face Detection in the Wild". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 8057–8066. DOI: 10.1109/CVPR42600.2020.00808.

[50] Nan Zhong et al. *PatchCraft: Exploring Texture Patch for Efficient AI-generated Image Detection*. 2024. arXiv: 2311.12397 [cs.CV].

[51] Jinghao Zhou et al. *iBOT: Image BERT Pre-Training with Online Tokenizer*. 2022. arXiv: 2111.07832 [cs.CV].

[52] Mingjian Zhu et al. *GenDet: Towards Good Generalizations for AI-Generated Image Detection*. 2023. arXiv: 2312.08880 [cs.CV].

[53] Mingjian Zhu et al. "GenImage: A Million-Scale Benchmark for Detecting AI-Generated Image". In: *Advances in Neural Information Processing Systems*. 2023, pp. 77771–77782. URL: https://proceedings.neurips.cc/paper_files/paper/2023/hash/f4d4a021f9051a6c18183b059117e8b5-Abstract-Datasets_and_Benchmarks.html.