Det-IGEN: Detectable Watermarking in Image Generation

CS-GY 6923 Project Presentation

Shashank Pandey (NetID: sp8108)
Sirish Parupudi (NetID: vsp7230)

PART 01

Problem Statement

Why is watermarking necessary?

Over the last two years, image generation models (OpenAI's DALL-E, Google's Gemini, xAI's Grok2) have become widely used.

In 2024, 39% of marketers in the United States used generative AI tools to create social media images.

These rapid advancements and the scale of adoption has raised a critical challenge:
the need for a reliable method to distinguish AI-generated images from authentic ones.

.

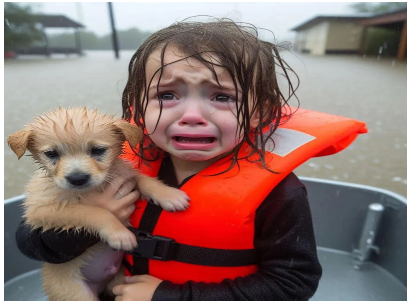
Craiyon
DALL·E Flow
Midjourney
Stable Diffusion
DALL·E 2
Black cat in a city made of yarn

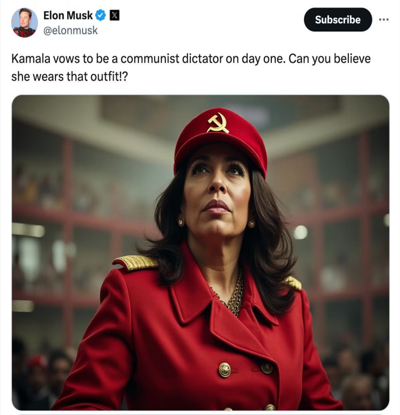
AI-Generated Misinformation

In the past few months, we have seen real-world examples of how AI-generated images can be seen as real.

1. During Hurricane Helene, **fake images of a tearful child holding a puppy in a rescue boat** went viral, misleading millions. These images undermined public trust in first responders and government, which hurt recovery efforts.

2. AI-generated images were also used to spread misinformation during the 2024 election: including images **showing Vice President Harris in communist attire**, and an image **showing pop star Taylor Swift endorsing former President Trump**.

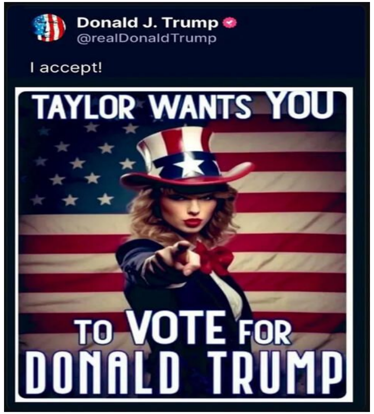




Elon Musk @elonmusk

Subscribe

Kamala vows to be a communist dictator on day one. Can you believe she wears that outfit!?



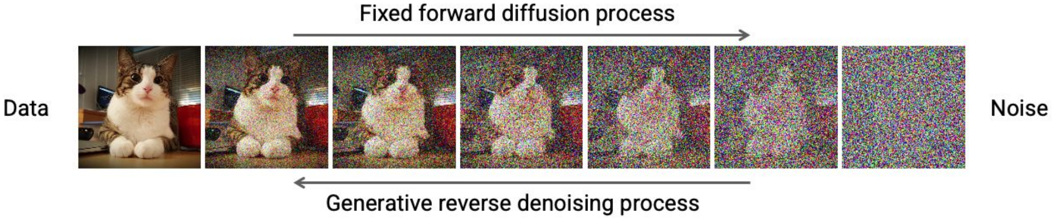
Donald J. Trump @realDonaldTrump

I accept!

Our Approach

In this project, we present Det-IGEN (**Det**ectable Watermarking in **I**mage **Gen**eration).

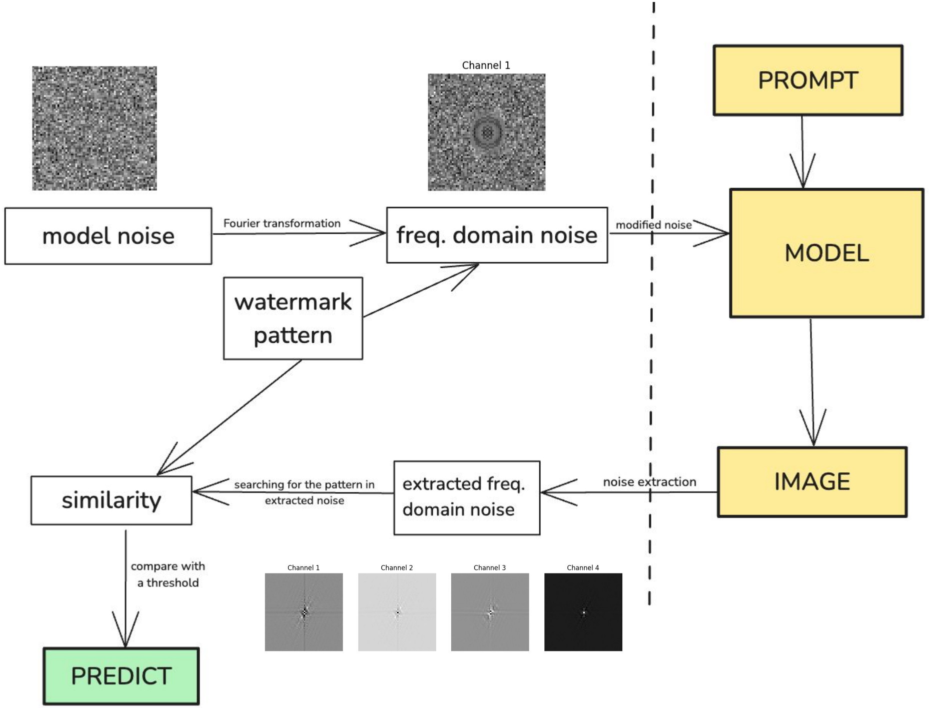
- Diffusion models contain components of random noise. These components are used to train the model: more and more noise is added to the image, and the model learns how to 'predict' the noise and get back the original image.


Fixed forward diffusion process
Data
Noise
Generative reverse denoising process

Our Approach

- In our approach, we augment this random noise, by adding a pattern into the frequency domain of this noise. The augmented model (with ‘watermarked’ noise) is now used to generate images.
- During detection - we effectively invert the generation process, by obtaining this random noise of the image in question.
- We then compute the distance between the Fourier transform of this noise and our watermark in the frequency domain, and judge it to be watermarked if the distance is below a threshold.
- We find that we get extremely promising results, performing well on both our tasks.

Pipeline for Tree-Ring Watermarking



“an airbrush painting of cyber war machine scene in area 5 1 by destiny womack, gregoire boonzaier, harrison fisher, richard dadd”

Watermarked Image 4

An, Bang, et al. "Benchmarking the robustness of image watermarks." *arXiv preprint arXiv:2401.08573* (2024).

Background Material

Image Generation Models

- Historically, **GAN's** (Generative Adversarial Networks) have been one of the most popular - and best-in-class approaches used to generate images. GANs comprise of 2 competing neural networks: a generator that creates synthetic data and a discriminator that distinguishes real from fake data.

- In recent years, **transformer-based** structures have also been explored as an alternative, where images are converted to token lists using an image decoder, which is our area of focus, especially with **diffusion models.**

Generative Image Detection: types of approaches

There are two broad approaches to generative image detection - passive detectors, and watermark-based detectors.

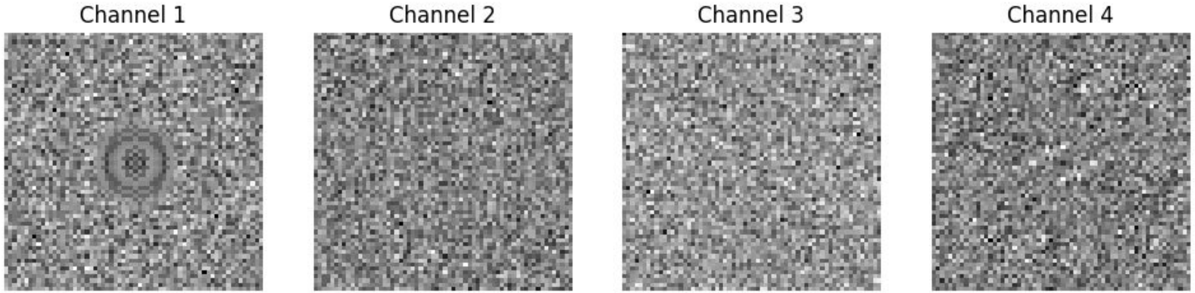
- In **passive detectors**, we use features from the image to classify images into AI-generated and non AI-generated classes. These features can include frequency domain information (as seen in FreqCLIP), or spatial information from the image (like in UnivCLIP). **These methods have shown to give sub-optimal results.**

- In **watermark-based approaches**, noise (a watermark) is added to some component in the generative process. This noise is what is leveraged/detected to make out that the image is in fact AI-generated. We are following a watermarking-based approach to design our model.

Watermarks

A watermarking method typically consists of three components: watermark, encoder, and decoder.

- The encoder embeds a watermark into content, while a decoder decodes a watermark from watermarked/unwatermarked content.
- When a content has a watermark, the decoded watermark is similar to our actual watermark. The encoder and watermark can also be embedded into the parameters of our model, such that its generated content is inherently watermarked.



Datasets

Prompts (for Task 1)

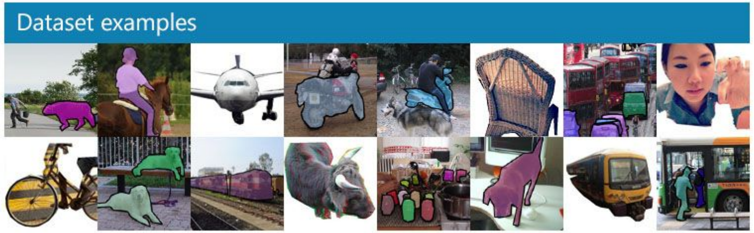
For the first task, we got prompts from the DiffusionDB dataset: huggingface.co/datasets/poloclub/diffusiondb

DiffusionDB includes 14 million detailed prompts for stable diffusion models, making it an ideal data source. It has also been used in multiple papers in the field of AI image detection.

Images (for Task 2)

For our second task, we got real images from the COCO image dataset: https://cocodataset.org/

COCO is a widely-used dataset, which contains real-world images across multiple different categories.



NYU

Preprocessing

1. **Resize**: Scales the image to the target_size (default is 512 pixels on both dimensions).
2. **Center Crop**: Crops the center of the image to ensure it is square with dimensions target_size x target_size.
3. **Convert to Tensor**: Converts the processed image into a PyTorch tensor.
4. **Normalize**: Adjusts the pixel values to the range [-1.0, 1.0] using the formula 2.0 * image - 1.0.

Solution Approach

Why Tree Rings?

The patterns can exploit several classical properties of the Fourier transform for periodic signals

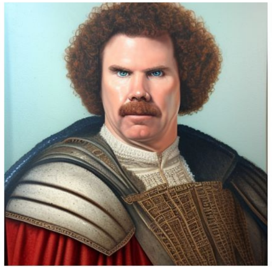
- A rotation in pixel space corresponds to a rotation in Fourier space.
- A translation in pixel space multiplies all Fourier coefficients by a constant complex number.
- A dilation/compression in pixel space corresponds to a compression/dilation in Fourier space.
- Color jitter in pixel space (adding a constant to all pixels in a channel) corresponds to changing the magnitude of the zero-frequency Fourier mode.

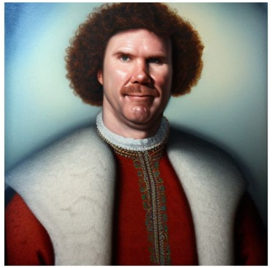
This makes it **highly resistant to image distortions and attacks**.

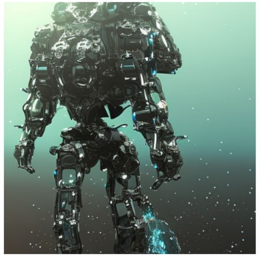
PART 04

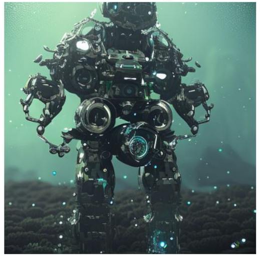
Watermarked vs Non-Watermarked

a renaissance portrait of will ferrell, art in the style of rembrandt!! intricate. ultra detailed, oil on canvas, wet - on - wet technique, pay attention to facial details, highly realistic, cinematic lightning, intricate textures, illusionistic detail,


Original Image 10

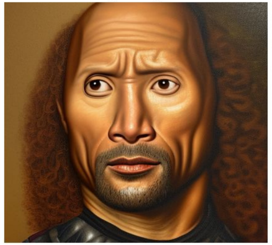

Watermarked Image 10

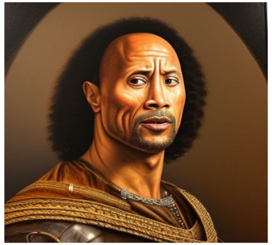

Original Image 3

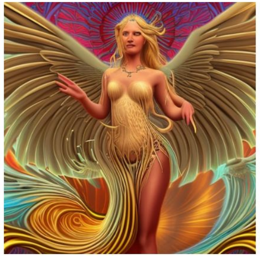

Watermarked Image 3

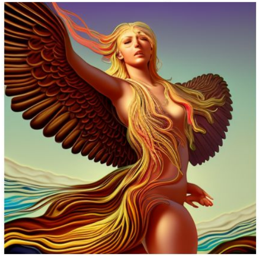
epic 3 d, become legend shiji! gpu mecha controlled by telepathic hackers, made of liquid, bubbles, crystals, and mangroves, houdini sidefx, perfect render, artstation trending, by jeremy mann, tsutomi nihei and ilya kuvshinov

a renaissance portrait of dwayne johnson, art in the style of rembrandt!! intricate. ultra detailed, oil on canvas, wet - on - wet technique, pay attention to facial details, highly realistic, cinematic lightning, intricate textures, illusionistic detail,


Original Image 1


Watermarked Image 1


Original Image 2

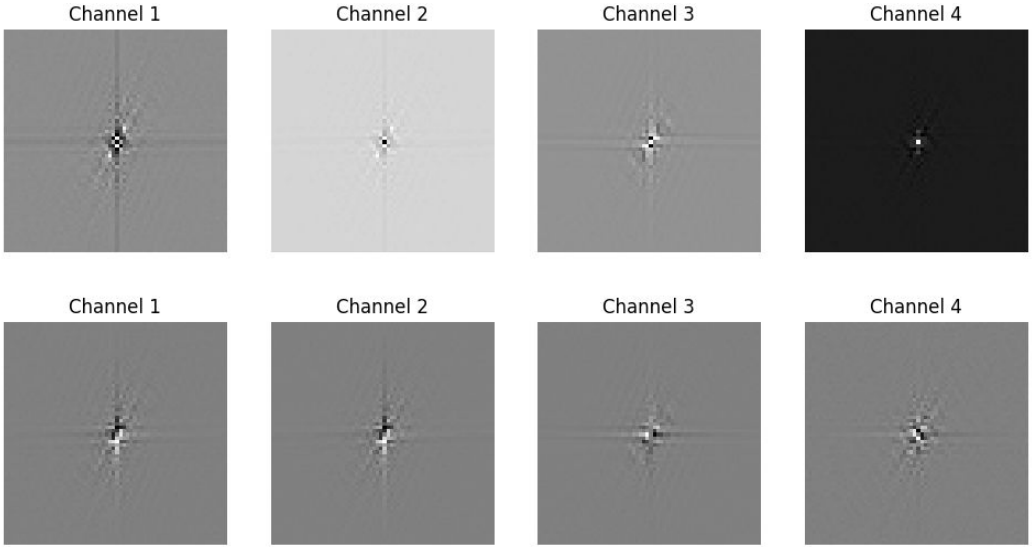

Watermarked Image 2

portrait of a dancing eagle woman, beautiful blonde haired lakota sioux goddess, intricate, highly detailed art by james jean, ray tracing, digital painting, artstation, concept art, smooth, sharp focus, illustration, artgerm and greg rutkowski and alphonse mucha, vladimir kush, giger, 8 k

Visualization of watermark in generated image

Original Image 4

Watermarked Image 4



Channel 1

Channel 2

Channel 3

Channel 4

Channel 1

Channel 2

Channel 3

Channel 4

Results & Analysis

To evaluate the model, we observe its performance in two tasks:

- In Task 1, we evaluate the performance of our detecting approach on a **mix of generated images, both with and without the watermark**. Here, ideally, we should be able to classify the images created into classes of: images with the watermark, and images without the watermark. This task aims to show that the **watermarks we embed** in the generation are being **successfully detected**.

- In Task 2, we evaluate the performance detecting approach on a **mix of watermark-generated images and real-world images**. This task is effectively the main goal of our project: being able to successfully and **consistently detect AI-generated images** will show that we have succeeded.

Since both our tasks are classification tasks, we use standard classification metrics like accuracy, precision, recall, and F-1 scores.

Results

We ran the model with different parameters: experimenting with the radius (size) of the watermark, along with the shape of the watermark. (we took the default radius as 10, and watermark shape as ‘random’)

Our results are as follows:

	Task 1: classifying watermarked images				Task 2: AI-generated image detection			
	accuracy	precision	recall	f1-score	accuracy	precision	recall	f1-score
radius = 10	0.99	0.99	1.0	0.99	0.99	0.99	1.0	0.99
radius = 5	0.99	0.99	1.0	0.99	1.0	1.0	1.0	1.00
ring-shaped watermark	0.96	0.93	1.0	0.97	0.98	0.96	1.0	0.98

Analysis

While going through the results, we observed some interesting insights:

- Across all models (and all tasks), not once did our detection algorithm incorrectly classify an watermarked AI-generated image as a real image (as we see that recall = 1.00 in all parameters). This is effectively an ‘**inescapable’** image detection paradigm.
- We are also encouraged by the **sparsity of false positives**. With precision scores > 0.9 in all runs, there are very few images that we are incorrectly classifying as AI-generated.
- Lastly, as shown in the outputs, the fact that the visible nature/**qualities of the images remain untouched** means that we can preserve the realism found in best-in-class image generation models, while still being able to watermark the process and detect it.

Challenges

Injection of watermarks through FFT

Performing watermark insertion into latent space proved tricky initially, but with the help of few fourier transformations into fourier space helped with watermark injection, followed by inverse fft to bring it back to normal space.

Had to perform inverse diffusion process as well to make sure our images indeed did have watermarks that were invisible to the naked eye.

Fine-tuning Diffusion Model

We had to stick to fine-tuning a pre-trained diffusion model, as our GPU resources were limited, we had to optimise with torch accelerate. We limited the number of prompts to 100 for generation and compiling our results, as higher numbers were not able to run on our notebooks.

Overcoming Loss of generation Quality

We modified watermark ring patterns, radius and other parameters like denoising steps to come to an optimal solution, in which the images generated with watermarks are not too far off from how the model normally generates images.

Real World Applications

- The images shown earlier were shared with millions of Americans by famous personalities like Elon Musk and former President Trump, without any indication that these images were fake and AI-generated.
- Stable Diffusion and Midjourney are two examples of contemporary text-to-image diffusion models that can produce a vast array of original images in an infinite number of styles. These systems are all-purpose picture generation technologies that can produce both realistic-looking phoney images and original artwork
- The creation of watermarks for text-to-image models' outputs is driven by the possibility of misuse. Through the identification of an image's origin, watermarks allow social media, news outlets, and the diffusion platforms themselves to limit harms or collaborate with law enforcement.

Conclusions

We have successfully implement a pipeline which can detect whether an image was AI generated or not. AI wrapper companies and deployable models can incorporate such methods to prevent the spread of misinformation and false propaganda.

Future Work

This problem can be extended to attribution and detection to trace back the user who generated the image. Further image encoding techniques with special messages can also be incorporated, for a more complex and advanced detection system.

References

1. Wen, Yuxin, et al. "Tree-ring watermarks: Fingerprints for diffusion images that are invisible and robust." arXiv preprint arXiv:2305.20030 (2023).
2. Fernandez, Pierre, et al. "The stable signature: Rooting watermarks in latent diffusion models." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023.
3. An, Bang, et al. "Benchmarking the robustness of image watermarks." arXiv preprint arXiv:2401.08573 (2024).

Thank you!

NYU