# Implementation of Digital Watermarks for Verifying AI-Generated Art

Felicia Sutandijo - 13520050
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): FeliciaSutandijo@gmail.com

***Abstract*—The rapid pace of artificial intelligence (AI) development has led to significant advancements in image generation, producing images and artworks that have started to be indistinguishable from those created by humans. This paper aims to address the critical need for verifying the origin and authenticity of AI-generated images through implementation of digital watermarks. Different watermarking techniques will be discussed and weighed in the context of watermarking AI-generated images. By evaluating these methods in terms of robustness, imperceptibility, and reliability, the paper aims to implement an algorithm suitable for watermarking AI-generated images. The approach utilizes DCT-based post-generation watermarking to balance the trade-offs, ensuring the credibility and traceability of digital art in the evolving AI landscape.**

***Keywords—DCT; watermark; AI image generation; blind watermarking; robust watermarking; colored image watermarking***

## I. Introduction

The rise of artificial intelligence (AI) has brought significant advancements in image generation, leading to an increase of AI-generated artworks. In an era where AI can create images that are indistinguishable from human-made art, verifying the origin and authenticity of these images becomes critical. One promising solution lies in the implementation of digital watermarking implemented in the pipeline of AI image generation. These watermarks aim to provide robust, imperceptible, and reliable methods to confirm not only that an image is AI-generated but also which specific AI model created it.

Fig. 1. AI generated image. Source: The New York Times

AI-generated images utilize algorithms such as neural style transfer and generative adversarial networks (GANs) to produce artworks. These images can be generated in a vast array of styles and forms, making them valuable assets in various fields, including media, marketing, and entertainment. However, distinguishing between human-made and AI-generated images poses a challenge, especially when considering the ease with which digital files can be copied and altered. Embedding digital watermarks into these images can serve as a means to authenticate the generation process and the specific AI model responsible, providing assurance of the image's AI origin.

Digital watermarking involves inserting information into a digital image in a way that is invisible to human observers but detectable through computational means. For AI-generated images, these watermarks must be designed to withstand common image alterations such as resizing, compression, and cropping while remaining undetectable during normal viewing. The watermark provides a digital signature that can be used to verify the source AI model, ensuring the integrity and authenticity of the image. This paper explores various techniques of embedding such watermarks into AI-generated images, comparing their effectiveness in terms of robustness, imperceptibility, and reliability in different scenarios. By doing so, it aims to implement an algorithm suitable for watermarking AI-generated images, ensuring their traceability and trustworthiness in the digital landscape.

## II. Theoretical Framework

### *A. AI Image Generation*

AI image generation refers to the use of artificial intelligence techniques to create visual content, often from textual prompts [1]. This field has seen significant advancements, driven by developments in machine learning, particularly deep learning. AI image generation includes a broad range of applications, including creating photorealistic images, artworks, and even entirely new scenes from textual descriptions or other inputs.

At its core, AI image generation leverages neural networks, which are computational models inspired by the human brain [1]. These networks are trained on vast datasets of images to

learn patterns and features that characterize different objects, textures, and scenes. Through this learning process, AI models develop the ability to generate new images that mimic the statistical properties of the training data, producing outputs that can be remarkably realistic or creatively abstract.

A fundamental aspect of AI image generation is the transformation of input data into visual outputs. This input can take various forms, such as random noise, text descriptions, or parts of images. The transformation process involves multiple layers of neural networks that progressively refine the image, adding details and improving coherence. This layered approach allows AI models to generate images that can range from simple shapes and patterns to complex, high-resolution scenes [1].

### *B. Digital Watermark*

Digital watermarking is a critical technique in information security and digital rights management that involves embedding imperceptible information into digital media to ensure authenticity, integrity, and ownership. This process embeds a hidden signal, known as a watermark, within digital content such that it remains undetectable under normal conditions but can be extracted or detected under specific circumstances. The watermark must be robust against various manipulations, including compression, cropping, and noise, ensuring it remains intact and retrievable. Achieving a balance between imperceptibility and robustness is crucial and guides the design of watermarking algorithms [2].

Watermarking techniques are broadly classified into spatial domain and frequency domain methods, each offering distinct advantages and limitations. Spatial domain techniques involve embedding the watermark directly into the pixel values of the image. A common example is the Least Significant Bit (LSB) modification, where the watermark is embedded in the least significant bits of the image pixels. This method is straightforward and computationally efficient but is more susceptible to common image processing operations, reducing its robustness [2].

In contrast, frequency domain techniques transform the image into a different domain before embedding the watermark. Methods such as the Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), and Discrete Fourier Transform (DFT) are commonly used. These techniques embed the watermark in the transformed coefficients, making the watermark more resilient to image manipulations like compression and filtering. For instance, in DCT-based watermarking, the image is divided into blocks, and the DCT is applied to each block. The watermark is then embedded in the mid-frequency coefficients, striking a balance between robustness and imperceptibility. These frequency domain techniques, while computationally more complex than spatial domain methods, offer enhanced robustness against various forms of manipulation [2].

Digital watermarking plays a crucial role in verifying AI-generated art, addressing the growing concern of distinguishing between human-created and AI-generated content. By embedding a unique, imperceptible watermark into AI-generated images, creators and platforms can ensure the authenticity and traceability of digital art. This embedded information can include details about the AI model used, the creation date, and the ownership, allowing for transparent provenance tracking and preventing unauthorized use or plagiarism. As AI art becomes more prevalent, digital watermarking provides a robust solution for maintaining the integrity and authenticity of artworks, facilitating trust and credibility in digital art marketplaces and among collectors, artists, and audiences.

### *C. DCT-Based Image Watermark*

DCT-based image watermarking is a widely used technique for embedding information within digital images, ensuring the security, authenticity, and traceability of the content. The Discrete Cosine Transform (DCT) is a mathematical transformation used in signal processing and image compression that converts spatial domain data into frequency domain data. In the context of watermarking, DCT enables the embedding of watermark information into specific frequency components of an image, making the watermark robust against various forms of image manipulation and compression [3].

The process of DCT-based watermarking typically involves several steps. First, the image is divided into non-overlapping blocks, usually of size 8x8 pixels. Each block is then transformed from the spatial domain to the frequency domain using the DCT. This transformation results in a set of DCT coefficients that represent the block's frequency components. The watermark information, often a binary sequence or a small image, is then embedded into the DCT coefficients of each block. To balance imperceptibility and robustness, the watermark is usually embedded in the mid-frequency coefficients, where changes are less likely to be noticed by the human eye but are still resilient to common image processing operations [3].

Once the watermark is embedded, the blocks are transformed back into the spatial domain using the inverse DCT (IDCT), resulting in the watermarked image. The embedded watermark can be extracted later by performing the DCT on the watermarked image, isolating the mid-frequency coefficients, and retrieving the embedded information. The extraction process often involves comparing the coefficients of the watermarked image with those of the original image or using a known pattern to detect the presence of the watermark [3].

DCT-based watermarking offers several advantages, particularly in terms of robustness and imperceptibility. Since the watermark is embedded in the frequency domain, it is less susceptible to common image processing operations such as compression, cropping, and noise addition. This makes DCT-based watermarking suitable for applications where the integrity of the watermark must be maintained despite potential modifications to the image. Additionally, because the changes to the image are made in the frequency domain, they are generally less perceptible to the human eye, preserving the visual quality of the watermarked image. However, the capacity of DCT-based watermarks to carry messages is limited usually only to mid-frequency coefficients, making it not ideal to carry detailed messages.

## III. Solution Analysis

### A. Use Cases

The primary use cases for digital watermarking in AI-generated imagery include enabling individuals to verify if an artwork is AI-generated and allowing platforms to distinguish AI-generated images from human-created ones. This distinction needs to be machine-readable, as traditional visual watermarks are often ineffective for automated detection systems.

### B. Requirements

The digital watermark must embed information that at a minimum confirms the image is AI-generated. Ideally, the watermark should also include data such as the specific AI model used, the creator of the image, the prompt involved in its generation, and other relevant metadata. The watermark must be robust, imperceptible to regular viewers, and reliable, remaining consistently present in any image generated by the AI model. Additionally, blind watermarking is essential, as verifiers will not always have access to the original image.

### C. Solution Alternatives

With the above requirements in mind, possible solutions to apply watermarking in AI generated images can be approached through two techniques, post-generation watermarking and embedded watermarking during generation.

Post-generation watermarking involves embedding the watermark after the image has been created. This method is straightforward and easy to implement. Depending on the algorithm chosen, it satisfies the requirements of being robust, imperceptible, and reliable. However, its simplicity makes it more vulnerable to removal by malicious entities. For example, removing post-generation watermarks in open-source models may be as simple as deleting one line of code.

Embedded watermarking during generation integrates the watermark during the image creation process. This can be achieved by training or fine-tuning an existing model to create a watermarking layer that is trained to embed a watermark with each image generation. This approach significantly increases the difficulty of removing the watermark, ensuring that each generated image carries it. While this method offers better security, it is more complex to execute. Aside from that, its reliability is not guaranteed, as AI may fail to insert a legitimate watermark on occasions, especially if training is not enough.

### D. Selected Solution

From the solution analysis, it is clear that several qualities are desired in the implementation of image watermarking for verifying AI-generated art. First of all, the type of signature must be a blind signature, for users do not have access to the original image when attempting to validate an AI-generated image. Second, it is important that the watermark be imperceptible, robust, and reliable. This leads to the choice of the implementation of a blind DCT-based watermark. In addition, for the sake of reliability (always present in every image generation) and time and resources constraints to train a machine learning model, the author has opted to implement a post-generation digital watermark solution. Lastly, it is essential that the watermarking function be implementable for colored images, since the majority of AI-generated images are colored images.

## IV. Implementation

### A. Parameters

This paper implements a highly customizable post-generation DCT-based watermark, which will enable fine-tuning the digital watermark function for different AI models according to their "art styles" in later stages. The parameters implemented are as follows.

- **Alpha**: An integer, ranging from 0 to 1, denoting the watermark strength. The closer it is to 0, the more imperceptible the watermark turns out in the final watermarked image. On the other hand, the closer it is to 1, the more robust the watermark becomes after being subject to several tests such as clipping, resizing, adding noise, and other attacks.
- **Beta**: An integer, ranging from 0 to 1, denoting the watermark size relative to the image size. Smaller numbers signify a worse quality watermark, possibly rendering it blurry (pixelated). However, smaller-sized watermarks creates a less noticeable change in the final watermarked image.
- **Channels**: A string with value of either "YCrCb", "YUV", or "RGB", denoting the channel splitting method of the function.
- **Channels list**: A list of integers, with values of 0, 1, and/or 2, denoting which channel to be embedded with the watermark. For example, if the channels parameter chosen is "YCrCb" and the channels list [0, 2], then the watermark will be embedded in the Y and Cb channels.

Fine-tuning the above parameters for a specific model yields the most compatible combination to ensure maximal imperceptibility and robustness for images generated by it.

### B. Function Implementation

#### 1) Setup

A DCT-based watermarking function is added to the image generation pipeline, right after an image is generated by the AI, and before it is served to the user.

#### 2) Preprocessing

Both the AI-generated image and the watermark image should both be converted into an array. Then, the watermark is resized by a factor of beta times the AI-generated image size. Lastly, the watermark should be converted to grayscale, if it is not yet in grayscale.

#### 3) Watermark Embedding

Watermark embedding begins by converting an image from the RGB space to the desired space (RGB, YCrCb, or YUV) according to the channels parameter. Next, DCT is performed

on the split channel according to the channels list parameter. If 0 is in the parameter, the R or Y channel is processed. Similarly, if 1 is in the parameter, the G or Cr or U parameter is processed. DCT operation is also performed on the gray-scaled watermark image.

After acquiring the required channels after DCT, the DCT values of the watermark is inserted to the selected channel(s). These values may be inserted at a certain offset, usually in the middle frequency, but the program used in this paper inserts from the back of the array of the original channel's DCT values. This ensures minimal disruption to the image to support imperceptibility. Furthermore, for the watermark to be extracted blindly, the proposed algorithm will not add the original DCT value to the weighted (by alpha) watermark DCT values, but rather only replace the original value with the weighted watermark DCT values. This method ensures that the watermark can be retrieved without the user having to possess the original image.

Following the watermark insertion, inverse DCT (IDCT) is performed to restore the image's channels to its values, with the addition of the watermark.

Lastly, all channels are merged back to create a whole colored image, similar to the original image, with the addition of the watermark in the selected channels. The watermark can then be extracted with the watermark extraction function.

*4) Watermark Extraction*

The watermark extraction is performed by reversing the embedding process and acquiring the watermark from the selected channel. To extract the watermark, the size of the watermark is needed to determine the size of the chunk to be extracted from the watermarked image. This is calculated by multiplying the size of the image with the beta factor.

The watermarked image is first split into three channels according to the channels parameter (RGB, YCrCb, or YUV). DCT is then performed on the selected channels, as the case in the watermark embedding function.

The watermark can be extracted from any channel that has been embedded with the watermark. This is done by extracting the values from the end of the DCT value arrays according to the size of the watermark image. In addition, watermark may also be extracted by averaging the values from all the channels that contain the watermark.

Inverse DCT (IDCT) is then performed to reverse the DCT and get the watermark image. There is no need to merge any channels like in the extract function as the watermark image has been converted to gray-scale and thus only contains one channel.

## V. EVALUATION

An image each from three different AI image generation models are tested with different parameters of the DCT-based function.

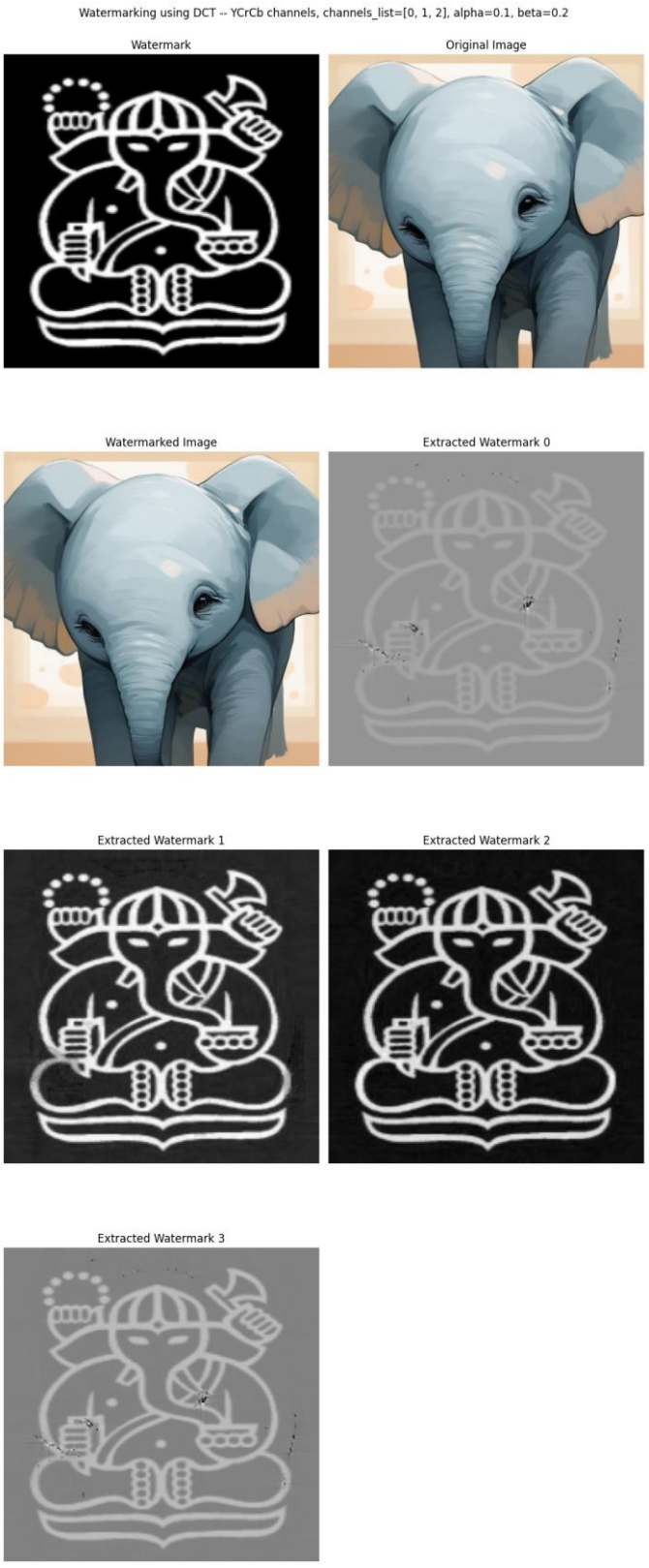


Fig. 2. AI image watermarking with DCT-based watermarks in YCrCb channels, image generated by the model Holodayo XL 2.1

In Figure 2, watermark embedding in different channels in the YCrCb channels is compared. The results show that

embedding a watermark in the Cr or Cb channel yields better results than embedding a watermark in the Y channel.

Below are comparisons between images from three different models with the same parameters, which are “YUV” for channels, [2] for channels list, 0.7 for alpha, and 0.4 for beta. The alpha and beta numbers are set higher to test the limits of robustness and imperceptibility for each model.

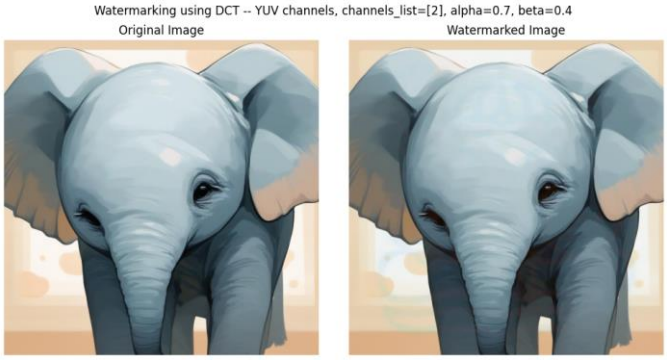


Fig. 3. Holodayo XL 2.1


Fig. 4. Stable Diffusion XL Base 1.0

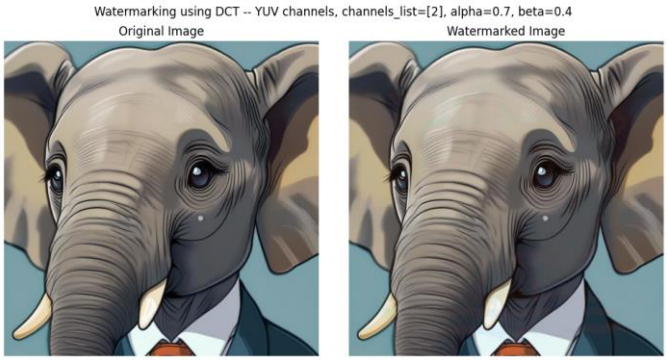


Fig. 5. Mobius

Notice that in Figure 3, the watermark can be seen overlaying the original image, especially on the white part of the background.

Lastly, the robustness of the watermarking method is tested with different attacks below.

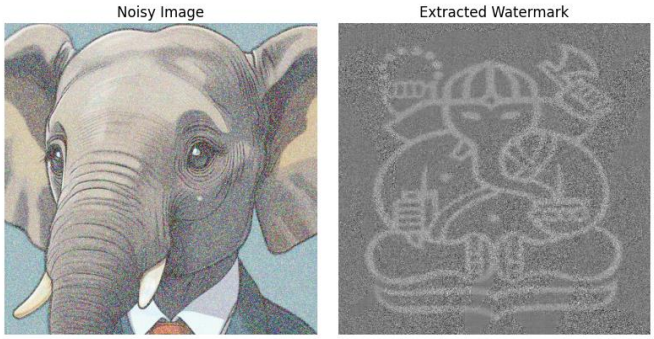


Fig. 6. Gaussian noise. Image generated by Mobius

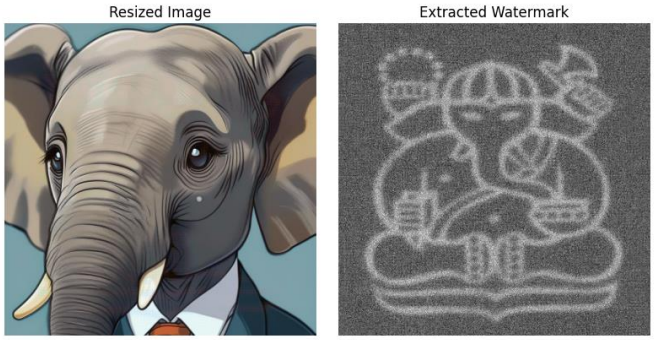


Fig. 7. Image resized by 0.5. Image generated by Mobius

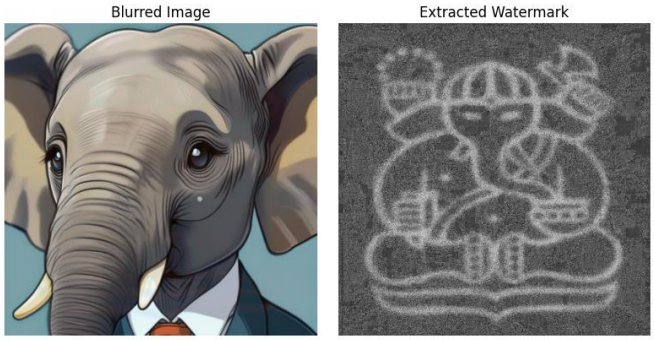


Fig. 8. Image blurred by ksize=(5, 5). Image generated by Mobius

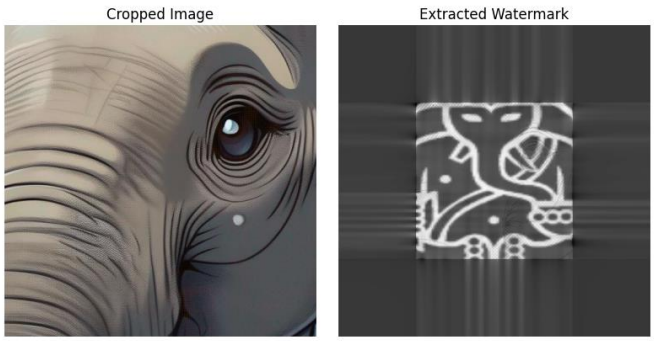


Fig. 9. Cropped image. Image generated by Mobius

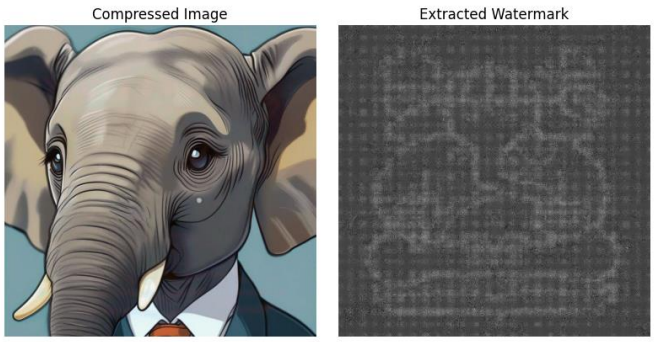


Fig. 10. Compressed image (JPG compression) by 70%. Image generated by Mobius

These tests show that the digital watermarking method can withstand various attacks, with the watermark able to be retrieved.

## VI. Conclusion

This paper implements a post-generation DCT-based blind digital watermark to verify AI-generated images, with the aim of maximizing robustness and imperceptibility. Configuring the right combination of parameters for the watermarking process depends on the each model's unique generation style. Different combinations serve different trade-offs to consider, especially in terms of robustness and imperceptibility.

The adoption of digital watermarking for AI-generated art ensures traceability, enhances trust in digital art markets, and helps combat unauthorized use and plagiarism. As AI continues to evolve, the methods discussed in this paper will be essential in maintaining the integrity and credibility of digital artworks.

## VII. Future Improvements

The implementation of digital watermarking to verify AI-generated art can further be improved by embedding the watermarking system into the generation process. This can be done by training or fine-tuning existing models to embed a suitable and unique watermark for each model reliably [4]. The function in this paper can be used to embed watermarks in the training data before it is fed to the model.

In terms of the function itself, improvements can be made by implementing an offset parameter for each channel. This parameter dictates where the start of the embedding of the watermark image is. This is especially useful if a certain model has a unique color distribution or style that makes inserting a watermark image at a certain offset result in a less perceptible watermark. In addition, varying the offsets for each channel has been proven to improve robustness dealing with various attacks [5].

Lastly, the precision of calculations used in this program can be improved to enable better embedding and extraction of watermark images. The program developed for this paper uses rounding before merging the three channels into the watermarked image. This may have caused poorer quality images due to rounding.

## GitHub Repository Link

The Python codes used in the experiments in this paper may be accessed from https://github.com/FelineJTD/digital-watermark-dct-ai-art.

## Acknowledgment

The author is thankful for the guidance and lessons from Dr. Ir. Rinaldi Munir, M.T. that have taught the basics of cryptography.

## References

[1] E. Enjellina, E. Vilgia Putri Beyan, and A. Gisela Cinintya Rossy, "A Review of AI Image Generator: Influences, Challenges, and Future Prospects for Architectural Field," *JARINA - Journal of Artificial Intelligence in Architecture*, vol. 2, Feb. 2023.

[2] R. Munir, *Kriptografi (Edisi Kedua)*. Penerbit Informatika.

[3] R. Munir, "Image Watermarking untuk Citra Berwarna dengan Metode Berbasis Korelasi dalam Ranah DCT," vol. 3, no. 1, 2010.

[4] P. Fernandez, G. Couairon, H. Jégou, M. Douze, and T. Furon, "The Stable Signature: Rooting Watermarks in Latent Diffusion Models." arXiv, Jul. 26, 2023. Accessed: Jun. 12, 2024. [Online]. Available: http://arxiv.org/abs/2303.15435

[5] I. El-Feghi, D. Mustafa, S. Zakaria, Z. Zubi, and F. El-Mouadib, "Color image watermarking based on the DCT-domain of three RGB color channels," Jan. 2009.

## PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 12 Juni 2024

Felicia Sutandijo
13520050