# An Effective Image Watermarking Scheme in YCbCr Color Space using 2-Level DWT

A PROJECT REPORT

Submitted by

Ankan Ghosh
Registration No.:223001810924
Roll No.:30017122003

Supervised by

Dr. Pabitra Pal

in partial fulfillment for the award of the degree

of

MASTER OF SCIENCE

IN

IT(ARTIFICIAL INTELLIGENCE)

Year: 2024


Dept of Information Science
School of Information Science and Technology
Maulana Abul Kalam Azad Univerity of Technology
Nadia, West Bengal, India

Dedicated to my Parents

# BONAFIDE CERTIFICATE

Certified that this project report "An Effective Image Watermarking Scheme in YCbCr Color Space using 2-Level DWT" is the bonafide work of "Ankan Ghosh" who carried out the project work under my supervision.

Signature of the HoD
Dr. Sayani Mondal
Assistant Professor, Dept of ET,
HoD, Dept of Computer Applications,
Maulana Abul Kalam Azad University of Technology

Signature of the Supervisor
Dr. Pabitra Pal
SUPERVISOR
Assistant Professor,
Dept of Computer Applications,
Maulana Abul Kalam Azad University of Technology

Signature of External Examiner

# Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degrees or diplomas of the university or other institutes of higher learning, except which due acknowledgment has been made in this text.

Signature of the Candidate

Ankan Ghosh

# Acknowledgments

I hereby wish to express my sincere gratitude and respect to Dr. Pabitra Pal, Dept. of Computer Applications, SIS&T, MAKAUT under whom I had proud privilege to work. His valuable guidance and encouragement have really led me to the path of completion of this project. Any amount of thanks would not be enough for the valuable guidance of my supervisor.

I would also like to thank all the faculty member of Dept of Information Science. for their devoted help. I also cordially thank all laboratory assistants for their cooperation.

Finally, I would like to pen down my gratitude towards my family members for their continuous support and encouragement. It would have not been possible to complete my work without their support.

# Abstract

This thesis introduces an innovative blind color image watermarking technique operating in the Y-Cb-Cr color space, utilizing color images for both the cover and watermark. The scheme employs a $2^{nd}$-level Discrete Wavelet Transform (DWT) to decompose the Y channel of the cover image. Watermark embedding occurs within the $HH_1$ sub-bands of the cover image, using a watermark strength ($\alpha$) of 0.1. Experimental analysis demonstrates the scheme's high imperceptibility, achieving a maximum PSNR of 45 dB for the watermarked image. The method exhibits robust performance under various attacks, notably maintaining a correlation coefficient (CC) of 0.98 under 95% JPEG compression. Remarkably, even at compression quality factors as low as 30%, the extracted watermark preserves all its color information. The proposed technique ensures efficient execution, with embedding and extraction times of 12.9 ms and 6.85 ms per image, respectively. Extensive testing across diverse datasets comprising over 6,000 images validates the method's effectiveness and reliability as a blind watermarking scheme.

Keywords: Blind Watermarking, RGB, YCbCr, DWT, JPEG Compression, color watermark

# Contents

# List of Figures

# List of Tables

# Abbreviations

- DWT: Discrete Wavelet Transform
- PSNR: Peak Signal-to-Noise Ratio
- SSIM: Structural Similarity Index Measure
- CC: Correlation Coefficient
- $\alpha$: Watermarking strength(0.1)
- RGB: Red Green Blue
- YCbCr: Luminance, Chrominance Blue, Chrominance Red
- SVD: Singular Value Decomposition
- RDWT: Redundant Discrete Wavelet Transform
- HVS: Human visual System
- IWT: Integer Wavelet Transform
- DCT: Discrete Cosine Transform
- ANN: Artificial Neural Networks
- LSVR: Lagrangian Support Vector Regression
- LWT: Lifting Wavelet Transform
- DTCWT: Dual-Tree Complex Wavelet Transform
- QDCT: Quaternion Discrete Cosine Transform
- GWO: Grey Wolf Optimizer

- DnCNN: Denoising Convolutional Neural Network
- LL1: Low-frequency approximation of the LL band of matrix
- LH1: Horizontal high-frequency details from LL band
- HL1: Vertical high-frequency details from LL band
- HH1: Diagonal high-frequency details from LL band
- LH: Horizontal high-frequency details of matrix
- HL: Vertical high-frequency details of matrix
- HH: Diagonal high-frequency details of matrix

# Chapter 1

# Introduction

## 1.1 Introduction

Digital watermarking has emerged as a crucial technology in the era of widespread digital content creation and distribution. A digital watermark is a subtle yet powerful tool for embedding valuable information directly into various forms of digital content, including images, audio files, and video files. Characterized by their inconspicuous nature, digital watermarks are carefully designed to avoid perceptibility, ensuring that they do not compromise the integrity or artistic value of the content.

The primary functions of digital watermarks revolve around safeguarding digital assets, verifying authenticity, and detecting potential tampering or alterations. By embedding a unique digital signature within the content, creators and owners can effectively protect their intellectual property rights and maintain control over their digital assets. Additionally, digital watermarks enable the tracking of content distribution and use, providing valuable insights into how digital assets are being utilized and shared.

The image watermarking, techniques have evolved to accommodate various color spaces and transform domains. The RGB (Red, Green, Blue) color model, widely used in digital imaging, presents both opportunities and challenges for watermarking. Converting RGB images to other color spaces, such as YCbCr, can offer advantages in terms of separating luminance and chrominance information, potentially leading to more robust watermarking schemes.

Blind watermarking, a subset of digital watermarking techniques, has gained significant attention due to its practical advantages. Unlike non-blind methods, blind watermarking allows for watermark extraction without the need for the original, unwatermarked content. This characteristic makes blind watermarking particularly suitable for real-world applications where access to the original content may be limited

or impractical.

The Discrete Wavelet Transform (DWT) has become a popular tool in image watermarking due to its ability to decompose an image into different frequency bands. This multi-resolution analysis allows for the embedding of watermark information in specific frequency components, potentially improving the trade-off between imperceptibility and robustness.

As digital content faces various forms of processing and compression in real-world scenarios, the robustness of watermarking schemes against such operations is crucial. JPEG compression, being one of the most common image compression techniques, poses a significant challenge to many watermarking methods. Therefore, developing watermarking techniques that can withstand JPEG compression while maintaining the integrity of the embedded information is of paramount importance.

This thesis aims to address these challenges by proposing a novel blind watermarking approach that leverages the YCbCr color space, DWT, and considers robustness against JPEG compression, specifically tailored for RGB images with RGB watermarks.

## 1.2 Blind watermarking

Blind watermarking marks a significant leap in digital watermarking, overcoming key limitations of traditional methods. Unlike conventional techniques that often require the original content or watermark for verification, blind watermarking operates independently of these elements. This approach is particularly well-suited for real-world applications. The hallmark of blind watermarking is its capacity to detect and extract watermarks using only the watermarked content, without needing the original image or watermark. This feature is vital in various scenarios, including:

- Content distribution platforms where original files are not readily available
- Digital rights management systems handling large volumes of media
- Forensic analysis of potentially tampered digital content
- Verification of content authenticity in social media and news dissemination

Blind watermarking techniques are designed to be robust against various forms of manipulation and processing, often encountered in digital media distribution and storage. These include compression, scaling, cropping, and various forms of intentional or unintentional attacks on the watermarked content.

In the context of this thesis, the focus is on blind watermarking for RGB images, utilizing the YCbCr color space and Discrete Wavelet Transform (DWT). This approach allows for the embedding and extraction of full-color watermarks, addressing the limitations of many existing methods that work primarily with grayscale or binary watermarks.

To facilitate a clear understanding of the concepts and methodologies discussed in this thesis, the following key terms are defined:

- Cover Image: The original digital image that will have a watermark embedded into it.
- Watermark Image: The smaller image or pattern that will be embedded into the cover image to identify ownership or authenticity.
- Watermarked Image: The resulting image after the watermark has been embedded into the cover image.
- Embedding: The process of inserting the watermark image into the cover image.
- Extraction: The process of detecting and retrieving the watermark image from the watermarked image.
- RGB Image: An image represented in the RGB color space, where each pixel is composed of three color values: Red, Green, and Blue.
- YCbCr Image: An image represented in the YCbCr color space, where each pixel is composed of three components: Luminance (Y) and two Chrominance values (Cb and Cr).
- DWT: This stands for Discrete Wavelet Transform. It's a mathematical algorithm used to decompose signals or images into different frequency components, representing them in a more compact and efficient way.
- Imperceptibility: The quality of a watermarked image where the embedded watermark is not noticeable or visible to the human eye(PSNR,SSIM,CC).
- 9. Attack: Any processing or manipulation of the watermarked image that may compromise the integrity of the embedded watermark, such as compression, cropping, or noise addition.

This research aims to advance the field of blind watermarking by proposing a novel technique that addresses the challenges of color preservation, robustness against JPEG compression, and efficiency in large-scale applications.

## 1.3 Problem Statement

Despite significant advancements in digital watermarking techniques, several critical gaps persist in the field, particularly in the context of blind watermarking for color images. This research aims to address the following key issues:

- Limited Dataset Utilization: Many existing studies rely on small or restricted datasets, limiting the generalizability and robustness of their findings. There is a pressing need for research that employs large, diverse datasets to validate watermarking techniques across a wide range of image types and qualities.

- Grayscale-Centric Approaches: A significant portion of current research focuses on grayscale images as cover media, neglecting the complexities and opportunities presented by color images. This limitation fails to address the full spectrum of real-world digital content, where color images are prevalent.

- Restricted Watermark Types: Most existing methods use binary or grayscale watermarks, which limit the amount and type of information that can be embedded. There is a lack of robust techniques for embedding and extracting full-color watermarks, which could potentially enhance security and information capacity.

- Inadequate Performance Evaluation: Many studies report execution times based on single-image processing, which does not provide a comprehensive understanding of the method's efficiency in real-world applications where bulk processing is often required.

- Color Integrity Post-Compression: A critical challenge lies in maintaining the color information of watermarks after common processing operations, particularly JPEG compression. Current methods often fail to preserve the full color characteristics of the embedded watermark after compression, compromising the watermark's integrity and recognizability.

This research proposes to address these gaps by developing a novel blind watermarking technique for RGB cover images that can embed and extract RGB watermarks. The proposed method aims to maintain color integrity even after JPEG compression, utilize large and diverse datasets for validation, and provide comprehensive performance metrics including bulk processing times. By addressing these issues, this study seeks to advance the field of digital watermarking, enhancing its applicability and effectiveness in protecting color image content in real-world scenarios.

## 1.4 Objectives

The primary goal of this research is to develop and evaluate a novel blind watermarking technique for RGB images that addresses the identified gaps in current watermarking methods. To achieve this, the following specific objectives have been formulated:

- Develop a robust blind watermarking algorithm: Design a method that uses RGB cover images and embeds RGB watermarks. Implement the algorithm using YCbCr color space conversion and 2-level DWT for enhanced performance.
- Ensure color integrity of the watermark: Develop techniques to maintain the watermark's information integrity specifically against JPEG compression.
- Optimize for imperceptibility and robustness: Achieve a high Peak Signal-to-Noise Ratio (PSNR) for watermarked images to ensure visual quality. Attain a high Correlation Coefficient (CC) for extracted watermarks to demonstrate robustness.
- Validate the method using extensive datasets: Test the proposed algorithm on a large and diverse dataset of over 6000 images to ensure generalizability and reliability.
- Evaluate computational efficiency: Measure and optimize the execution time for both embedding and extraction processes. Assess performance in bulk processing scenarios to demonstrate real-world applicability.
- Analyze robustness against JPEG compression: Evaluate the method's performance under various levels of JPEG compression, with a focus on maintaining watermark integrity at 95% compression.
- Compare with existing methods: Benchmark the proposed technique against state-of-the-art watermarking methods to demonstrate its advantages and improvements
- Provide comprehensive performance metrics: Report detailed results including PSNR, CC values, execution times, and performance under various image processing operations.

By achieving these objectives, this research aims to contribute a significant advancement to the field of digital image watermarking, offering a more robust, efficient, and widely applicable method for protecting color image content in real-world scenarios.

## 1.5 Significance of the Study

This research contributes significantly to the field of digital image watermarking, addressing several critical gaps in current methodologies and offering practical solutions for real-world applications. The significance of this study can be outlined as follows:

- Advancing Digital Rights Management: This study's novel approach to blind watermarking of RGB images contributes to more effective protection of digital content, potentially revolutionizing how digital rights are managed in the age of widespread image sharing and manipulation.
- Enhancing Data Security: By improving the robustness of color watermarks against JPEG compression, this research addresses a critical vulnerability in current image security methods, offering stronger protection for sensitive visual information.
- Improving User Experience: The high PSNR achieved by this method ensures that watermarking doesn't degrade image quality, allowing for widespread adoption without compromising visual aesthetics.
- Facilitating Forensic Analysis: The ability to embed and extract full-color watermarks provides richer information for digital forensics, potentially aiding in the detection of image tampering or unauthorized use.
- Expanding Industrial Applications: By demonstrating efficiency in bulk processing, this study opens up new possibilities for large-scale implementation in industries like photography, media, and e-commerce.
- Contributing to Standardization Efforts: The comprehensive evaluation using a large dataset contributes valuable data that could inform future watermarking standards and best practices.
- Inspiring Interdisciplinary Research: This work at the intersection of image processing, color theory, and information security may spark new avenues of research in related fields.
- Addressing Real-World Challenges: By focusing on blind watermarking of color images, this study directly addresses practical challenges faced in content creation, distribution, and authentication pipelines.

This research's significance lies not just in solving technical problems, but in its potential to impact how digital images are secured, shared, and authenticated in an increasingly visual digital world. It provides a foundation for more secure and efficient handling of color image content across various platforms and industries.

## 1.6 Thesis Organization

This thesis is structured into five chapters, each focusing on specific aspects of the research. The organization is as follows:

Chapter 1: Introduction This chapter provides an overview of the research, including the background, problem statement, objectives, and significance of the study. It also introduces key concepts in blind watermarking and outlines the thesis structure.

Chapter 2: Literature Review This chapter presents a comprehensive review of existing literature in the field of digital image watermarking. It covers:

- Analysis of 20 existing research papers on watermarking techniques
- Color space transformations, including detailed matrices for RGB to YCbCr conversion and vice versa
- Discrete Wavelet Transform (DWT), with a focus on 2nd level DWT
- Embedding and extraction equations used in proposed watermarking method
- Matrix operations, including the use of slicing operators for splitting matrices into even and odd components,with reshaping the watermark image
- Evaluation metrics for watermarking techniques

Chapter 3: Proposed Methodology This chapter details the novel aspects of the proposed blind watermarking method for RGB images. It includes:

- Detailed explanation of the watermark embedding and extraction algorithm
- flow charts of embedding and extraction process
- algorithm of the embedding and extraction process

Chapter 4: Experimental Results and Analysis This chapter presents the experimental setup, results, and analysis of the proposed method. It covers:

- Description of the dataset used for evaluation
- Performance metrics: PSNR,SSIM, Correlation Coefficient, execution time
- Analysis of robustness against JPEG compression,different brightness,and other attacks such as cropping, blurring, and scaling
- Comparative analysis with existing method

- Discussion of results and their implications

Chapter 5: Conclusion and Future Work This final chapter summarizes the key findings and contributions of the research. It also discusses:

- Limitation of the proposed method
- Directions for future research in this field

References: A comprehensive list of all sources cited throughout the thesis.

This organization ensures a logical flow of information, from introducing the research problem to presenting the solution, evaluating its performance, and concluding with insights and future directions. Each chapter builds upon the previous one, providing a comprehensive and coherent presentation of the research work.

# Chapter 2

# Literature Review and Theoretical Background

## 2.1 Review of Existing Watermarking Techniques

This section examines 20 significant research papers that have contributed to the field of digital image watermarking over the past two decades. These studies provide a comprehensive overview of current methodologies, challenges, and advancements in the area.

In 2018, Ernawan and Kabir [1] proposed a hybrid blind watermarking scheme that integrates RDWT with SVD and uses entropy measure for the cover image and Arnold chaotic map for the binary watermark. Testing reveals better robustness compared to existing methods.

In 2018, Bei et al. [2] addressed the limitations of traditional image watermarking algorithms, characterized by poor robustness and susceptibility to geometric attacks. Their proposed color-blind watermarking algorithm combines DWT and DCT to enhance robustness. converts the image from RGB TO YCbCr format, and the watermark is embedded in all three color spaces. The uses Human Visual System (HVS), along with DCT and DWT, and uses Zernike moments for rotation correction. Testing shows robustness against geometric attacks.

In 2021, Sinhal et al. [3] proposed a blind color image watermarking technique using the YCbCr color space, IWT, and DCT. To enhance the security of the watermark, randomized blocks which are dependent on the secret key. The scheme employs an ANN architecture to reduce computational complexity. Testing shows the scheme's effectiveness in terms of imperceptibility and robustness.

In 2010, Goléa et al. [4] proposed an RGB image watermarking technique using SVD. Their paper is the only one that uses color images for both cover and wa-

termark. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2015, Mehta et al. [5] proposed an image watermarking scheme that efficiently combines LSVR with LWT to balance imperceptibility and robustness, concurrently reducing time complexity Security is improved through Arnold transformation applied to the original watermark, with scrambled bits embedded by comparing LSVR outputs with actual target values. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2023, Teoh et al. [6] proposed an SVD-based image watermarking scheme, they showed that such algorithms offer a good trade-off between robustness and imperceptibility but they face vulnerability to false-positive issues. To address this, they used a combination of SVD, HVS, DWT. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2015, Benoraira et al. [7] proposed a blind image watermarking technique using DWT and DCT, they used 256 binary data as watermark. The scheme uses a simple but effective embedding and extraction equation, it is based on the assumption that neighboring pixels will have almost similar values. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2011, Makhloghi et al. [8] suggested a blind image watermarking technique, using SVD and DWT. The embedding process involves modifying SVD from a cover image with the bits obtained from the SVD of the watermark image. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2015, Roy et al. [9] developed a non-blind image watermarking technique in the YCbCr color space using DWT and SVD. The paper challenges the existing norm by asserting that YCbCr. The effectiveness of the Lewis-Barni HVS watermarking model is tested in this color space. The proposed scheme leverages Arnold scrambling for watermark security, Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2019, Tan et al. [10] proposed a non-blind watermarking technique using YCbCr, DWT, and SVD. The embedding is done in the HL subband of 4th level DWT. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2018, Alzahrani and Memon [11] suggested a Blind image watermarking technique using DWT, DCT, SVD, and human visual system (for selecting regions of interest). Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2022, Asikuzzaman et al. [12] developed a video watermarking technique, particularly in the face of camcording attacks using DTCWT and SVD. Current solutions struggle to embed imperceptible watermarks resistant to distortions introduced by camcording. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2014, Kakkirala and Chalamala [13] proposed a blind image watermarking technique using torus automorphism, DWT, and SVD. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2007, Yin et al. [14] developed a watermarking technique where embedding is done in the green channel of an image using DWT and SVD. The green component undergoes decomposition into LLn, HLn, LHn, and HHn subbands using DWT at the Nth levels. Different methods are adopted for watermark embedding in each subband. For LLn, pseudo-random embedding is performed after spreading based on energy. The three wavelet matrix coefficients LHn, HLn, and HHn are decomposed using SVD. The embedding process involves adding the SVD of the watermark and the wavelet coefficients. Retrieval and blind detection algorithms are designed accordingly. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2010, Naderahmadian and Hosseini-Khayat [15] proposed a blind watermarking technique using QR decomposition and DWT. The proposed scheme exhibits low computational complexity. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2023, Yang et al. [16] propose a deep blind watermarking method to enhance the imperceptibility and robustness trade-off in traitor tracing for copyright protection. The researchers also present a watermarking strategy that increases capacity with reduced training time and VRAM usage, Testing shows the technique is robust against various signal processing attacks.

In 2021, Hsu and Hu [17] propose a quaternion discrete cosine transform (QDCT)-based watermarking scheme using GWO and DnCNN. Binary embedding, tailored to the attributes of each QDCT component, is optimized using GWO, while DnCNN enhances the visual recognizability of extracted binary watermarks. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2019, Khedr and Elsoud [18] developed a blind watermarking technique for embedding 3 gray images in a color image, utilizing DWT, DCT, and SVD. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2010, Chen and Chen [19] proposed a blind watermarking technique using BPNN.

The trained models capture specific relationships used in the embedding process. Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

In 2019, Huynh-The et al. [20] proposed a blind image watermarking technique designed to effectively combat intentional attacks on watermark destruction. The approach employs a deep convolutional encoder-decoder network, allowing the system to learn attacking patterns. A binary watermark image is concealed within selective wavelet blocks using an optimal encoding rule to minimize image quality degradation and enhance imperceptibility. The framework then reveals embedding maps, representing wavelet coefficient differences, from various simulated attacks on the watermarked image. These maps are used to train a deep learning-based watermark extraction model, enabling precise recovery of the hidden watermark information from an attacked image.Experimental evaluations demonstrate that this method remains effective when subjected to diverse forms of signal manipulation and interference.

This extensive review highlights several key trends and developments in the field of digital watermarking:

- Evolution of transform domains: Many techniques utilize various transform domains such as DWT, DCT, SVD, and more recently, QDCT and DTCWT.
- Color space utilization: While many techniques focus on grayscale images, there's a growing trend towards utilizing color spaces, particularly YCbCr.
- Integration of machine learning: Recent studies have incorporated artificial neural networks, deep learning, and other ML techniques to enhance watermarking performance.
- Robustness against attacks: A consistent focus across studies is improving robustness against various signal processing and geometric attacks.
- Blind vs. non-blind techniques: While both approaches are represented, there's a trend towards developing blind watermarking techniques for their practical advantages.
- Imperceptibility-robustness trade-off: Many studies aim to balance these two crucial aspects of watermarking.
- Security enhancements: Several techniques incorporate additional security measures, such as Arnold transformation or torus automorphism.

This comprehensive review forms the foundation for understanding the current state of the art in digital watermarking and highlights the gaps that the proposed method aims to address. The next section will present a comparative analysis of these techniques to further elucidate the strengths and limitations of existing approaches.

## 2.2 Comparative Analysis of Existing Techniques

To synthesize the findings from the literature review, table 2.1 summarizes the key aspects of the reviewed watermarking techniques.

Table 2.1: Literature Review

| Scheme | TECHNIQUES USED | ACHIEVEMENT | Dataset |
|---|---|---|---|
| [1] 2018 | redundant wavelet transform, SVD, Arnold chaotic map | NC: 0.9875 for JPEG QF of 40 | 6 grayscale images of SIPI dataset, of size $512 \times 512$, |
| [2] 2018 | DCT HVS DWT, conversion to YCbCr, Zernike moments | NC: 0.98 and 0.96 for rotation of 45 degree | color image from SIPI dataset of size 512×512 |
| [3] 2021 | DCT IWT ANN | Embed time:0.4 sec with ann and 1.9 sec without ann | 80 color images of size $512 \times 512$ |
| [4] 2010 | SVD | RGB cover and watermark image, PSNR: 32 to 50dB | color image from sipi dataset of sizes 256x256, 512x512, 1024x1024 |
| [5] 2015 | LSVR, LWT, Arnold transform | Execution time of 1.4second | images from sipi dataset of size $512 \times 512$ |
| [6] 2023 | Arnold transformation, AES-192 encryption, DWT, SVD, HVS | robust against many different attacks, max watermark size of $90 \times 90$ | 3 color image from sipi dataset of size $512 \times 512$ |
| [7] 2015 | DWT, DCT | PSNR between 42 and 46. | GRAY SCALE image from sipi dataset of size $512 \times 512$ |
| [[8] 2011 | DWT, SVD | PSNR between 62 and 63. | 4 grayscale images from sipi dataset, of size $512 \times 512$, |
| [9] 2015 | DWT, SVD, cHVS, conversion to YCbCr, Arnold transform | NC:0.964 for rotation of 20 degree, PSNR:51dB | color images from sipi dataset |

Continued on next page

Table 2.1 – Continued from previous page

| Scheme | TECHNIQUES USED | ACHIEVEMENT | Dataset |
|---|---|---|---|
| [10] 2019 | DWT, SVD, conversion to YCbCr, Arnold transform | NC:0.9998 for rotation of 20 degree, PSNR:55dB | 15 color images of size between 500 × 480 and 768 × 512 |
| [11] 2018 | SVD, DWT, DCT, HVS | NC: 1 for JPEG QF of 50 | 16 grayscale medical images of size 1024 × 1024, |
| [12] 2022 | DTCWT, SVD | Embedding :0.262s Extraction :0.126s | 10 grayscale images of size 1920 × 1080, |
| [13] 2014 | DWT, SVD, Torus Automorphism | BER:0.29 for JPEG compression(lossy) | Image of size 512 × 512 |
| [14] 2007 | DWT, SVD, Chaos Scrambles | NC: 0.8775 for JPEG QF of 30 | Color image of Baboon of size 200 × 200 |
| [15] 2010 | Arnold transformation, QR Decomposition, DWT | NC: 0.9879 for JPEG QF of 37.5 Execution time:146s | 10 grayscale images of size 512 × 512 |
| [16] 2023 | Deep neural network | Embedding :0.001s Extraction :0.003s Lesser vram usage | 10,000 color images of coco dataset of size 256 × 256 |
| [17] 2021 | GWO, DnCNN, QDCT | NC: 0.973 for JPEG QF of 40 | 64 color images of size 512 × 512 |
| [18] 2019 | SVD, DWT, DCT | NC: 0.90 to 0.98 for rotation of 90 degree | color image of lena of size 512 × 512 |
| [19] 2010 | BPNN, ECC, DWT, Arnold scrambling, chaotic sequence | BER: 0 for JPEG QF of 90∼25 | 100 grayscale images of size 512 × 512 |
| [20] 2019 | DWT, Deep convolutional encoder-decoder network | NC: 0.999 for JPEG QF of 10 And average NC of 0.995 for a variety of robustness test | 10000 grayscale images from BOSSbase1.01 database of size 512×512 |

This comparative analysis, in conjunction with the detailed literature review, reveals several important trends and gaps in the current state of digital image watermarking:

1. Limited datasets: Many studies rely on small or restricted datasets, which may limit the generalizability of their findings.

2. Grayscale-centric approaches: While some techniques use color images as cover, there's a predominant focus on grayscale images for both cover and watermark, neglecting the complexities and opportunities presented by full-color watermarking.

3. Inadequate performance evaluation: Most studies report execution times based on single-image processing, which doesn't provide a comprehensive understanding of the method's efficiency in real-world applications where bulk processing is often required.

4. Color integrity post-compression: There's a critical challenge in maintaining the color information of watermarks after common processing operations, particularly JPEG compression. Many current methods fail to preserve the full color characteristics of the embedded watermark after compression.

5. Trade-off between imperceptibility and robustness: Studies often prioritize one over the other, with few achieving a balanced approach that maintains high image quality while ensuring watermark resilience.

6. Limited exploration of full-color watermarks: Few techniques address the embedding and extraction of full-color watermarks, which could potentially enhance security and information capacity.

These observations highlight the need for a more comprehensive approach to digital image watermarking. The proposed method aims to address these gaps by developing a technique that:

- Utilizes large and diverse datasets for validation
- Focuses on full-color watermarking for both cover and watermark images
- Maintains color integrity even after JPEG compression
- Provides comprehensive performance metrics including bulk processing times
- Strives for a balance between imperceptibility and robustness

By addressing these issues, This thesis seek to advance the field of digital watermarking, enhancing its applicability and effectiveness in protecting color image content in real-world scenarios.

## 2.3 Color Space Transformations

### 2.3.1 YCbCr Color Space Conversion

The conversion of images from the RGB format to the YCbCr format plays a crucial role in the watermarking process. The YCbCr color space is preferred for its ability to separate luminance (Y) from chrominance (Cr and Cb), offering advantages in image compression and processing as also shown by Roy et al. [9]. In this application, the Y channel is primarily utilized for watermark embedding and extraction due to its importance in representing image intensity. The Y channel provides robustness against JPEG compression. JPEG compression, commonly used for reducing the file size of images, can introduce artifacts that may affect watermark visibility and robustness. Focusing on the Y channel, which contains luminance information and is less affected by chrominance changes, ensures that the watermark remains perceptually invisible and robust against compression. The conversion from RGB to YCbCr was achieved using the following matrix:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.168736 & -0.331264 & 0.5 \\ 0.5 & -0.418688 & -0.081312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.1}$$

For the reverse conversion from YCbCr back to RGB, the following matrix was utilized:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 1.402 \\ 1.0 & -0.344136 & -0.714136 \\ 1.0 & 1.772 & 0.0 \end{bmatrix} \begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} \tag{2.2}$$

### 2.3.2 Advantages of YCbCr in Watermarking

1. Separation of Luminance and Chrominance: This separation allows for more targeted watermark embedding, typically in the luminance channel, which is less perceptible to the human eye.

2. Compatibility with JPEG: YCbCr is the native color space used in JPEG compression, making it advantageous for watermarking techniques that need to be robust against JPEG compression.

3. Perceptual Uniformity: The YCbCr space is more perceptually uniform than RGB, allowing for better control over the visual impact of the watermark.

4. Efficient Processing: Working in YCbCr can lead to more efficient processing, as changes can be made to the luminance channel without affecting color information.

### 2.3.3 Considerations in Implementation

When implementing YCbCr conversion in watermarking algorithms, it's important to consider:

1. Precision: Ensure that the conversion is performed with sufficient numerical precision to avoid introducing artifacts.
2. Range Adjustment: YCbCr values may need to be adjusted to fall within the standard range (16-235 for Y, 16-240 for Cb and Cr in 8-bit systems).
3. Reversibility: Ensure that the conversion process is fully reversible to maintain image quality after watermark extraction.

The watermarking technique leverages the YCbCr color space, specifically the Y (luminance) channel, to maintain the watermark's information integrity against JPEG compression.

## 2.4 Discrete Wavelet Transform in Watermarking

The Discrete Wavelet Transform (DWT) is a powerful tool used in digital image watermarking to identify regions of interest for embedding and extracting watermarks. DWT offers a multi-resolution analysis of the image, allowing for robust watermarking techniques. This thesis specifically utilize the 2nd level DWT on the Y channel of the image.

### 2.4.1 2-level DWT Decomposition

The 2nd level DWT decomposes the image into following subbands: LL1, LH1, HL1, and HH1, LH, HL, HH. Each of these subbands represents different frequency components of the image:

- LL1: Low-frequency approximation of the LL band of matrix
- LH1: Horizontal high-frequency details from LL band
- HL1: Vertical high-frequency details from LL band
- HH1: Diagonal high-frequency details from LL band
- LH: Horizontal high-frequency details of matrix
- HL: Vertical high-frequency details of matrix

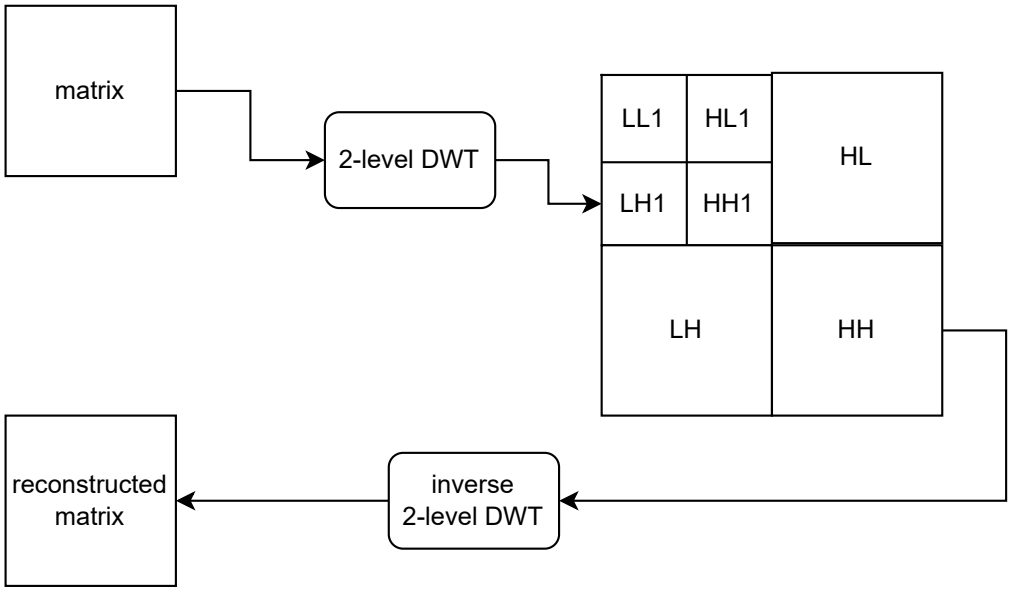


Figure 2.1: 2-level DWT decomposition and reconstruction

- HH: Diagonal high-frequency details of matrix

To illustrate the 2D DWT decomposition, the subbands can be represent as figure:2.1

### 2.4.2 HH1 Subband for Watermarking

For the watermarking process, the HH1 band is focused upon. This band is obtained by performing DWT on the LL band resulting from the first level DWT of the original matrix. In essence, a 2nd level DWT is applied to the matrix to extract the HH1 band, as illustrated in the figure 2.1. The HH1 band is selected for several reasons. It typically contains high-frequency information and is well-suited for watermark embedding due to its sensitivity to small changes. Moreover, this subband offers a unique combination of robustness and imperceptibility. The robustness is inherited from the LL band of the 1st level DWT, while the imperceptibility is achieved through the 2nd level DWT in the HH1 band extracted from the LL band.

### 2.4.3 Advantages of DWT in Watermarking

Utilizing the DWT for watermarking offers several advantages:

1. Multi-resolution analysis: Allows for embedding at different scales and frequencies.

2. Localization: Provides both spatial and frequency domain localization.

3. Human Visual System (HVS) compatibility: The frequency separation aligns well with the HVS, allowing for more imperceptible watermarking.

4. Robustness: DWT-based watermarks are generally more robust against common image processing operations.

5. Compression friendliness: DWT is the basis for many image compression standards, making it suitable for compression-resistant watermarking.

### 2.4.4 Implementation Considerations

When implementing DWT-based watermarking, it's important to consider:

1. Choice of wavelet: Different wavelet families (e.g., Haar, Daubechies) can affect the performance of the watermarking scheme.

2. Level of decomposition: Higher levels of decomposition can offer more robustness but may affect imperceptibility.

3. Coefficient selection: Careful selection of coefficients for modification is crucial for balancing robustness and imperceptibility.

## 2.5 Matrix Manipulation Techniques

The thesis employ several matrix manipulation techniques to effectively embed and extract the watermark. These techniques involve splitting and merging matrices, reshaping the watermark image, and applying specific embedding and extraction equations.

### 2.5.1 Splitting and Merging Matrices

To prepare the cover image for watermark embedding, the HH1 subband of the Y channel is split into even and odd element matrices. This process involves using array slicing techniques to separate alternate rows of the matrix.

- For splitting: create two separate matrices: one containing the even-indexed rows and another containing the odd-indexed rows of the original matrix.

- For merging: reconstruct the original matrix by alternately inserting the rows from the even and odd matrices back into their original positions.

This splitting and merging process allows for more nuanced manipulation of the cover image during watermark embedding and extraction.

### 2.5.2 Watermark Image Reshaping

To effectively embed the watermark into the cover image, reshape the 3D watermark image into a 2D matrix. This process involves:

1. Analyzing the dimensions of the original watermark image.
2. Reshaping the 3D image array into a 2D matrix, where the third dimension (color channels) is incorporated into the first two dimensions.

For extraction, perform the reverse process:

1. Taking the extracted 2D watermark matrix.
2. Reshaping it back into a 3D array based on the original watermark dimensions.

This reshaping process allows to maintain the full color information of the watermark while adapting it to the 2D nature of the embedding process.

## 2.6 Embedding and Extraction Equations

In the watermarking process, embedding involves hiding the watermark information in the cover image, while extraction retrieves the watermark from the watermarked image. The equations employed in the insertion and retrieval processes play a vital role in safeguarding the durability and authenticity of the concealed data.

### 2.6.1 Embedding Equations

The embedding process involves two matrices, A and B, representing the cover image, and a watermark matrix C. The embedding equations are as follows:

$$x = \frac{A + B}{2} \tag{2.3}$$

$$A' = x + C \tag{2.4}$$

$$B' = x - C \tag{2.5}$$

### 2.6.2 Extraction Equation

For the extraction process, the watermarked matrices A' and B' is needed to calculate C, the extraction equation is:

$$2C = A' - B' \tag{2.6}$$

These equations form the core of the proposed watermarking algorithm, enabling effective embedding and accurate extraction of the watermark. By leveraging these matrix manipulation techniques and equations, the watermarking method simultaneously preserves the visual quality of the cover image and enhances the watermark's resilience against digital manipulation and compression. This dual achievement is accomplished while maintaining the full color information of both the cover image and the watermark, resulting in a robust yet visually seamless watermarking solution.

## 2.7 Evaluation Metrics for Watermarking

To assess the performance of the watermarking technique, three key metrics are employed : Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and Correlation Coefficient (CC). These metrics provide a comprehensive evaluation of the watermarking method's imperceptibility and robustness.

### 2.7.1 Peak Signal-to-Noise Ratio (PSNR)

PSNR is used to measure the quality of the watermarked image compared to the original cover image. It is expressed in decibels (dB) and is defined as:

$$PSNR = 10 \log_{10} \left( \frac{MAX_I^2}{MSE} \right) \tag{2.7}$$

where $MAX_I$ is the maximum possible pixel value of the image (255 for 8-bit images), and MSE is the Mean Squared Error:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2 \tag{2.8}$$

Here, $I$ represents the original image, $K$ the watermarked image, and $m$ and $n$ are the image dimensions.

### 2.7.2 Structural Similarity Index (SSIM)

SSIM provides a more perceptual evaluation of image quality. It compares local patterns of pixel intensities across the original and watermarked images. The SSIM is defined as:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{2.9}$$

where $\mu_x$ and $\mu_y$ are the average pixel values, $\sigma_x^2$ and $\sigma_y^2$ are the variances, $\sigma_{xy}$ is the covariance, and $c_1$ and $c_2$ are constants to stabilize division with weak denominators.

### 2.7.3 Correlation Coefficient (CC)

The Correlation Coefficient measures the similarity between the original and extracted watermarks. for its implementation, the proposed method utilize NumPy's corrcoef function, which calculates the Pearson correlation coefficient. The mathematical definition of CC is:

$$CC = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \tag{2.10}$$

where $X$ and $Y$ are the original and extracted watermarks respectively, $\bar{X}$ and $\bar{Y}$ are their means, and $n$ is the number of pixels. The corrcoef function computes this efficiently for 2D arrays, returning a correlation matrix. This thesis uses the off-diagonal element of this matrix, which represents the correlation between the original and extracted watermarks. These metrics provide a comprehensive evaluation of the watermarking technique:

PSNR quantifies the imperceptibility of the watermark in the cover image. SSIM offers a perceptual measure of image quality after watermarking. CC assesses the robustness of the watermark by measuring how well it can be extracted after potential attacks or transformations.

By using these metrics in combination, they can effectively evaluate both the imperceptibility and robustness of their watermarking method, ensuring a balance between these crucial aspects of watermarking performance.

## 2.8 Robustness Analysis: Attacking Cases

To evaluate the robustness of the watermarking technique, the watermarked images is subjected to various attacks that simulate real-world scenarios of image manipulation. After each attack, the method extracts the watermark and calculate the Correlation Coefficient (CC) values across all datasets, recording both the maximum and mean

values for each dataset. This comprehensive approach allows to assess the resilience of the method under diverse conditions.

### 2.8.1 JPEG Compression

JPEG compression is one of the most common operations performed on digital images. Many existing techniques struggle to maintain color information in the extracted watermark after JPEG compression, especially when the watermark is embedded in the RGB channels. The technique proposed in this research incorporates hidden data into the luminance component of the YCbCr color model, with the goal of maintaining chromatic fidelity even when subjected to significant compression.

tested the robustness against JPEG compression at various quality factors: 30%, 40%, 50%, 60%, 70%, 80%, and 95%

This range allows to evaluate performance under extreme compression (30%) to near-lossless compression (95%).

### 2.8.2 Brightness Attack

Adjusting image brightness is a common image processing operation. A test was conducted to assess the watermark's resilience to brightness variations, including both brightening and darkening, at the following levels: 50%, 60%, 70%, 80%, 90%, 110%, 120%, 130%, 140%, 150%

This range covers significant darkening (50%) to notable brightening (150%) of the image.

### 2.8.3 Cropping Attacks

To simulate scenarios where malicious actors remove portions of the image, cropping attacks were employed. Two types of cropping were tested:

Cropped upper left quadrant (25%): Removes a quarter of the image from the upper left corner. Cropped border (25%): Removes a border around the entire image, reducing its size by 25

### 2.8.4 Scaling Attacks

Scaling attacks simulate image resizing operations. Tested for two scenarios:

Down Scale (100→75→100): The image is first reduced to 75% of its original size, then scaled back up to 100%. Up Scale (100→150→100): The image is first enlarged to 150% of its original size, then scaled back down to 100%.

### 2.8.5 Rotation Attack

The watermarked image was subjected to a slight rotation of 0.3 degrees. This subtle manipulation, while imperceptible to the human eye, can significantly alter the pixel values.

### 2.8.6 Gaussian Blur

A Gaussian blur with a 3x3 kernel was applied to the watermarked image. This attack simulates loss of sharpness that can occur in image processing or transmission.

### 2.8.7 Salt and Pepper Noise

added salt and pepper noise with a density of 0.001 to the watermarked image. This attack simulates impulse noise that can occur due to errors in image transmission or sensor malfunction.

For each of these attacks, we:

1. Applied the attack to the watermarked cover image
2. Retrieved the embedded watermark image from the manipulated watermarked image
3. Calculated the CC values between the original and extracted watermarks across all datasets
4. Recorded the maximum and mean CC values for each dataset

This comprehensive analysis allows for the evaluation of the watermarking method's robustness against a wide range of potential manipulations. By maintaining high CC values across these diverse attacks, the resilience and effectiveness of the technique in preserving watermark integrity under various real-world scenarios can be demonstrated.

# Chapter 3

# Proposed Methodology

Building upon the theoretical foundations and insights gained from the literature review, this chapter presents the core of the research. The method addresses the gaps identified in existing approaches, particularly focusing on maintaining color integrity in both the cover image and the watermark, even under various attacks and transformations.

The proposed methodology consists of two main components: the watermark embedding process and the watermark extraction process. These processes are designed to work in tandem, ensuring that the watermark can be effectively hidden within the cover image and accurately recovered when needed.

## 3.1 Watermark Embedding Process

The embedding process involves calling the algorithm described in Section 3.1, which internally refers to the embedding process algorithm detailed in Section 2. The corresponding flow charts for these processes are depicted in Figures 3.1 and 3.2.

To begin, the cover image is first converted to the YCbCr format. Subsequently, the Y channel undergoes a second-level discrete wavelet transform (DWT). All DWT coefficients are extracted, from which the HH1 band is selected for embedding the watermark data.

Next, the watermark image is converted into a 2D matrix by stacking all its color channels. This matrix is then multiplied by the watermarking strength $\alpha$ to compress its information. The shape of this compressed watermark matrix is calculated for further use.

The HH1 band is then split into two matrices containing odd and even pixel values using the slicing operator. The middle portion of both matrices, which is in the same shape as the compressed watermark matrix, is extracted.As per Equation 2.3

the adjacent pixel values are averaged to obtain an intermediate 2D matrix, and two copies of it are created.

Two copies of the compressed watermark matrix are made.As per the Equation 2.4 One copy is added to one copy of the intermediate 2D matrix to obtain a new even middle portion matrix, while as per the Equation 2.5 the other copy is subtracted from the other copy of the intermediate 2D matrix to obtain a new odd middle portion matrix.

Next, the new even and odd middle portions are replaced in their respective even and odd pixel location matrices. These matrices are then combined back to obtain the new HH1 matrix, which now contains the watermark information.

This new HH1 matrix replaces the original HH1 matrix in the DWT coefficients. The inverse second-level DWT is then performed to obtain the new Y channel, which replaces the original Y channel. Finally, the resulting YCbCr image is converted back into RGB format to obtain the watermarked image.

Algorithm 1 EMBEDDING

Input: cover image, watermark image
Output: watermarked image

1: YCbCr_image ← RGB_To_YCbCr(cover image).
2: Y, Cb, Cr ← Split_channels(YCbCr_image)
3: coefficient_DWT ← 2nd_level_DWT(Y).
4: LH1, HL1, HH1 ← coefficient_DWT[1].
5: waterRow,waterColumn,waterDepth ← watermark image.shape()
6: 2DwaterRow←waterRow×waterDepth
7: 2DwaterColumn←waterColumn
8: 2D_watermark_matrix←watermark image.reshape(2DwaterRow,2DwaterColumn)
9: newHH1 = EMBEDDING PROCESS(HH1,2D_watermark_matrix) (See Algorithm 2).
10: coefficient_DWT[1] ← LH1, HL1, newHH1
11: newY ← 2nd_level_DWT_inverse(coefficient_DWT)
12: newYCbCr_image = combine_channels(newY, Cb, Cr)
13: watermarked image ← YCbCr_To_RGB(newYCbCr_image)
14: Return watermarked image

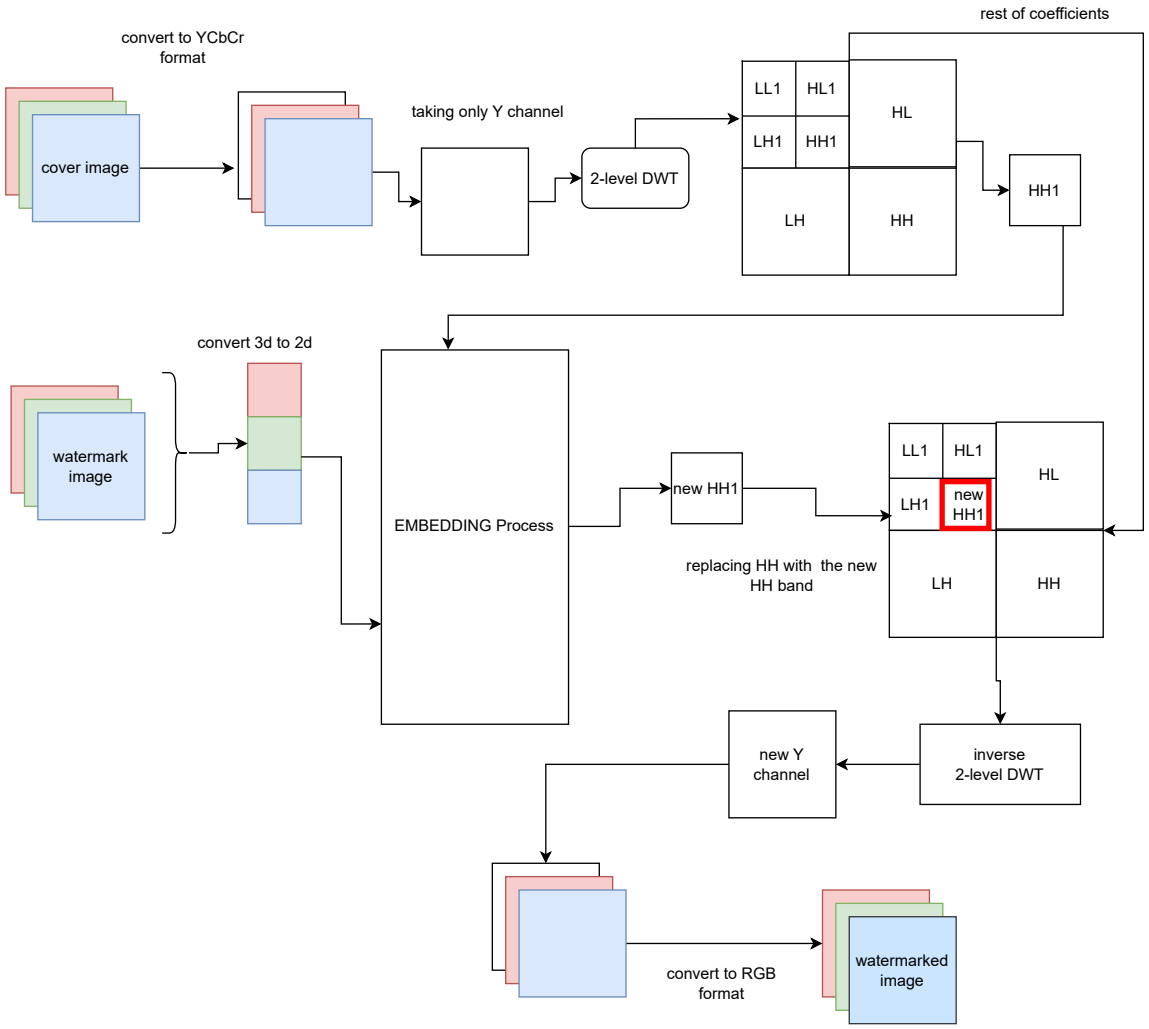


Figure 3.1: Block diagram of overall Embedding Process

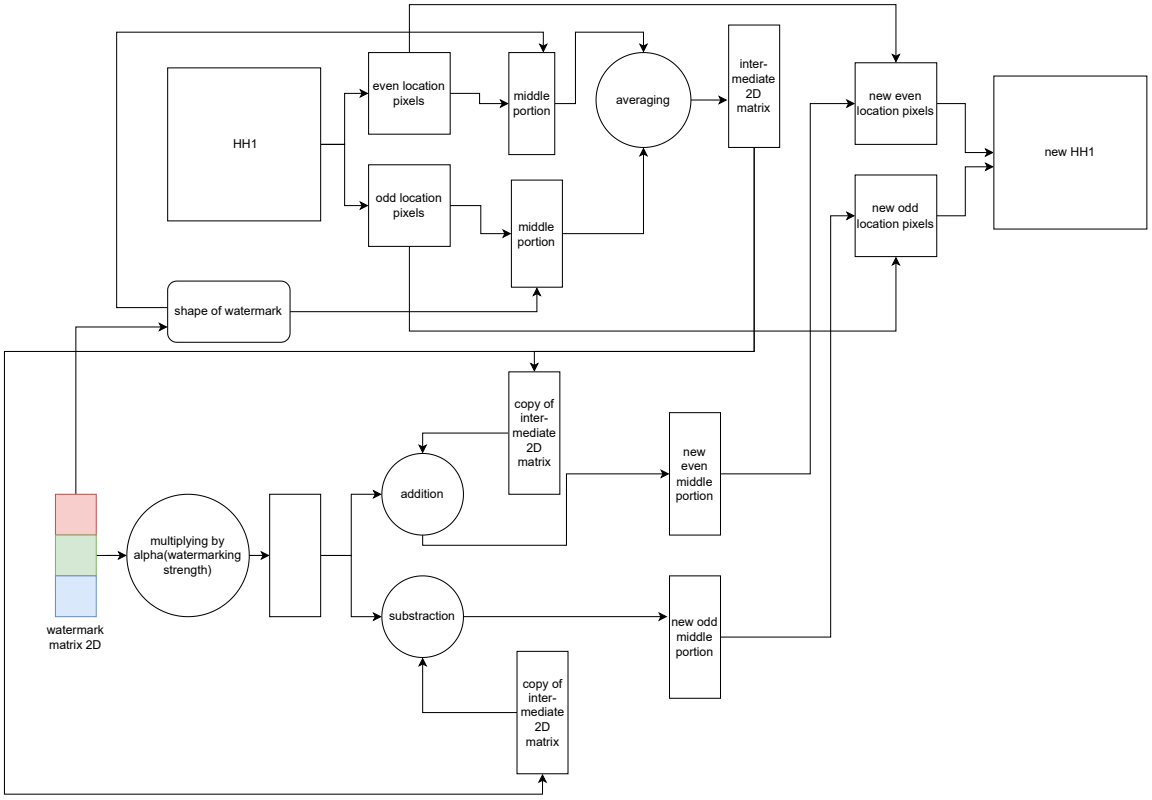


Figure 3.2: Detailed diagram of Embedding Process

Algorithm 2 EMBEDDING PROCESS

Input: HH1,2D watermark matrix

Output: newHH1

2DwaterRow,2DwaterColumn←2D watermark matrix.shape()
compressed_watermark_matrix ← 2D watermark matrix × $\alpha$(here $\alpha$ taken as 0.1)
newHH1 = HH1.copy()
EVEN← HH1[::2,:]
ODD← HH1[1::2,:]
EVENrow,EVENcolumn = EVEN.shape()
ODDrow,ODDcolumn = ODD.shape()
partEVENrow←(EVENrow−2DwaterRow)÷2
partEVENcolumn←(EVENcolumn−2DwaterColumn)÷2
partEVEN←EVEN[partEVENrow : partEVENrow+2DwaterRow, partEVENcolumn : partEVENcolumn+2DwaterColumn]
partODDrow←(ODDrow−2DwaterRow)÷2
partODDcolumn←(ODDcolumn−2DwaterColumn)÷2
partODD←ODD[partODDrow : partODDrow+2DwaterRow, partODDcolumn : partODDcolumn+2DwaterColumn]
newpartEVEN←((partEVEN+partODD)÷2)+compressed_watermark_matrix
newpartODD←((partEVEN+partODD)÷2)−compressed_watermark_matrix
newEVEN←(EVEN[partEVENrow : partEVENrow+2DwaterRow, partEVENcolumn : partEVENcolumn+2DwaterColumn] = newpartEVEN)
newODD←(ODD[partODDrow : partODDrow+2DwaterRow, partODDcolumn : partODDcolumn+2DwaterColumn] = newpartODD)
newHH1[::2,:]←newEVEN
newHH1[1::2,:]←newODD
Return newHH1

## 3.2 Watermark Extraction Process

The extraction process involves calling the algorithm described in Section 3.2, which internally refers to the extraction process algorithm detailed in Section 4. The corresponding flow charts for these processes are depicted in Figures 3.3 and 3.4.

To begin, the watermarked image is converted into the YCbCr format. The Y channel is then selected, which undergoes a second-level discrete wavelet transform (DWT). All DWT coefficients are extracted, and the HH1 band is selected.

Next, the HH1 band is split into two matrices containing odd and even pixel values. The middle portion of both matrices is extracted based on the inputted value, which corresponds to the shape of the watermark.

Subsequently, subtraction is performed between the odd and even pixel middle portion matrices. The result is then divided by 2 and again by the watermark strength $\alpha$, as shown in Equation 2.6, to obtain the required decompressed 2D watermark matrix.

This decompressed watermark matrix is unstacked, dividing it into three equal portions according to its number of rows, and then converted into a single 3D matrix to obtain the extracted RGB watermark image.

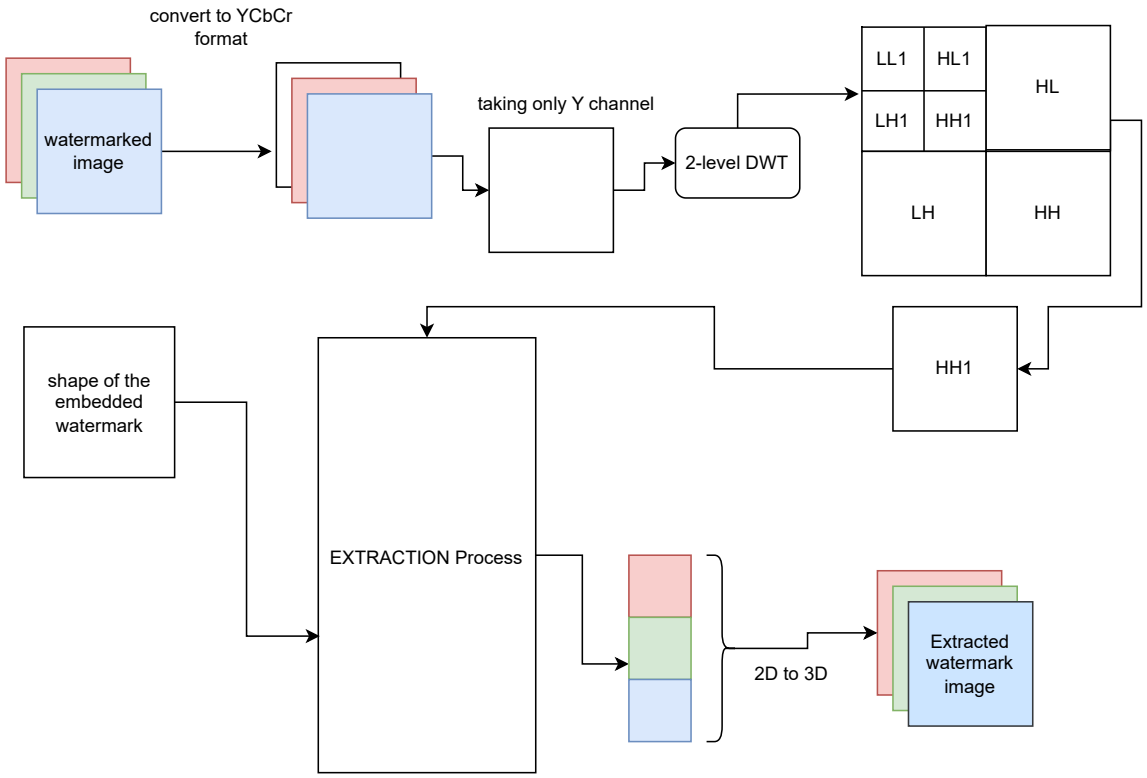


Figure 3.3: Block Diagram of Extraction Scheme

Algorithm 3 EXTRACTION

Input: Watermarked image, Shape of watermark image

Output: Extracted watermark image

1: YCbCr_image ← RGB_To_YCbCr(Watermarked image).
2: Y, Cb, Cr ← Split_channels(YCbCr_image)
3: coefficient_DWT ← 2nd_level_DWT(Y).
4: LH1, HL1, HH1 ← coefficient_DWT[1].
5: 2D_watermark_matrix = EXTRACTION PROCESS(HH1, shape of watermark image) (See Algorithm 4).
6: watermark image ← 2D_watermark_matrix.reshape(Shape of watermark image)
7: Return watermark image

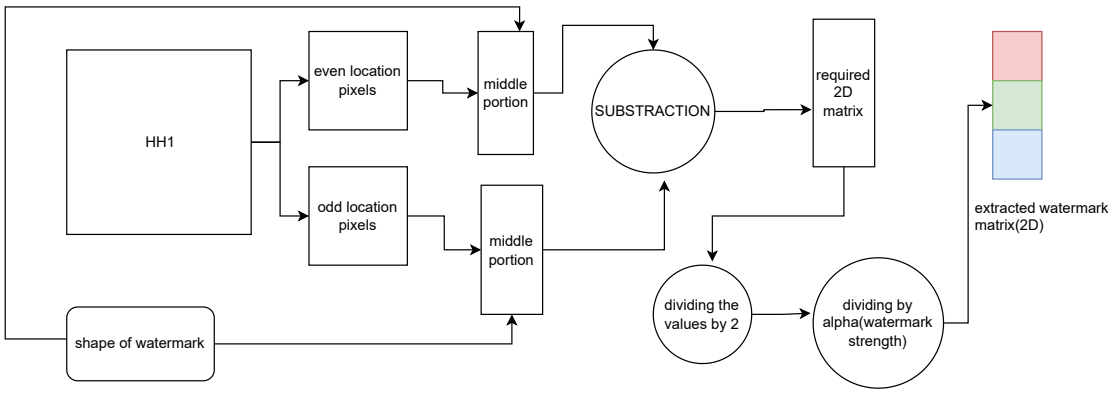


Figure 3.4: Detailed process for extraction scheme

Algorithm 4 EXTRACTION PROCESS

Input: HH1,shape of watermark matrix

Output: 2D watermark matrix

1: waterRow,waterColumn,waterDepth ← shape of watermark matrix
2: 2DwaterRow←waterRow×waterDepth
3: 2DwaterColumn←waterColumn
4: shape_watermark_2D_matrix ← (2DwaterRow,2DwaterColumn)
5: EVEN← HH1[::2,:]
6: ODD← HH1[1::2,:]
7: EVENrow,EVENcolumn = EVEN.shape()
8: ODDrow,ODDcolumn = ODD.shape()
9: partEVENrow←(EVENrow−2DwaterRow)÷2
10: partEVENcolumn←(EVENcolumn−2DwaterColumn)÷2
11: partEVEN←EVEN[partEVENrow : partEVENrow+2DwaterRow, partEVENcolumn : partEVENcolumn+2DwaterColumn]
12: partODDrow←(ODDrow−2DwaterRow)÷2
13: partODDcolumn←(ODDcolumn−2DwaterColumn)÷2
14: partODD←ODD[partODDrow : partODDrow+2DwaterRow, partODDcolumn : partODDcolumn+2DwaterColumn]
15: 2D watermark matrix ← ((partEVEN − partODD)÷ 2)÷ $\alpha$(here $\alpha$ taken as 0.1)
16: Return 2D watermark matrix

By the end of this chapter, readers will have a comprehensive understanding of the proposed watermarking technique, its innovative aspects, and how it addresses the challenges identified in current watermarking methods.

# Chapter 4

# Results and Analysis

## 4.1 Experimental Setup

The experiments were conducted on a dataset of more than 6000 RGB images of $512 \times 512$ resolution, sourced from the SIPI database and Kaggle's "130k Images ($512 \times 512$) - Universal Image Embeddings." This dataset comprises a variety of image types, including artwork, cars, dishes, furniture, illustration, and packaged images.

The watermark used in the experiments is a custom RGB image with a resolution of $32 \times 32$. The watermarking strength parameter $\alpha$ was set to 0.1.

The experiments were performed on a computer with the following specifications: AMD RYZEN 7 4800H processor (8 cores/16 threads) running at 4.2GHz, 16GB RAM, and a 512GB PCIe Gen.3 SSD. Python 3.10.13 was used for implementation.

## 4.2 Performance Metrics

The performance of the proposed watermarking algorithm was evaluated using three key metrics: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and Correlation Coefficient (CC). These metrics were calculated by comparing the watermarked images with their corresponding cover images across various datasets. Table4.1 presents the performance metrics for watermarked images from different datasets: The algorithm achieved impressive results across all datasets:

1. Peak PSNR: 45.232019 dB (Dishes dataset)
2. Maximum SSIM: 0.995833 (Furniture dataset)
3. Maximum CC: 0.999873 (Dishes dataset)

Table 4.1: performance metric of watermarked images of Various Datasets

| Dataset | Metric | Maximum | Mean |
|---|---|---|---|
| SIPI | PSNR | 40.862977 | 39.297493 |
| | SSIM | 0.983252 | 0.973974 |
| | CC | 0.999381 | 0.997915 |
| Artwork | PSNR | 45.219366 | 39.595862 |
| | SSIM | 0.994369 | 0.972160 |
| | CC | 0.999842 | 0.998754 |
| Cars | PSNR | 42.427692 | 39.781509 |
| | SSIM | 0.986489 | 0.971175 |
| | CC | 0.999792 | 0.999220 |
| Dishes | PSNR | 45.232019 | 39.196338 |
| | SSIM | 0.991429 | 0.976539 |
| | CC | 0.999873 | 0.999113 |
| Furniture | PSNR | 43.838766 | 40.418265 |
| | SSIM | 0.995833 | 0.968246 |
| | CC | 0.999825 | 0.998873 |
| Illustrations | PSNR | 42.444661 | 39.254377 |
| | SSIM | 0.991190 | 0.968063 |
| | CC | 0.999820 | 0.998679 |
| Packaged | PSNR | 41.642119 | 39.559359 |
| | SSIM | 0.990000 | 0.975731 |
| | CC | 0.999642 | 0.999018 |

These values indicate that the algorithm maintains high image quality across diverse image types. The mean values across datasets are also consistently high, with PSNR ranging from 39.196338 to 40.418265 dB, SSIM from 0.968063 to 0.976539, and CC from 0.997915 to 0.999220.

Figure 4.1 displays sample images from the SIPI database, including original cover images, watermarked images, and extracted watermark images. Visual inspection of these images confirms the algorithm's ability to preserve image quality, as the differences between the original and watermarked images are imperceptible to the naked eye.

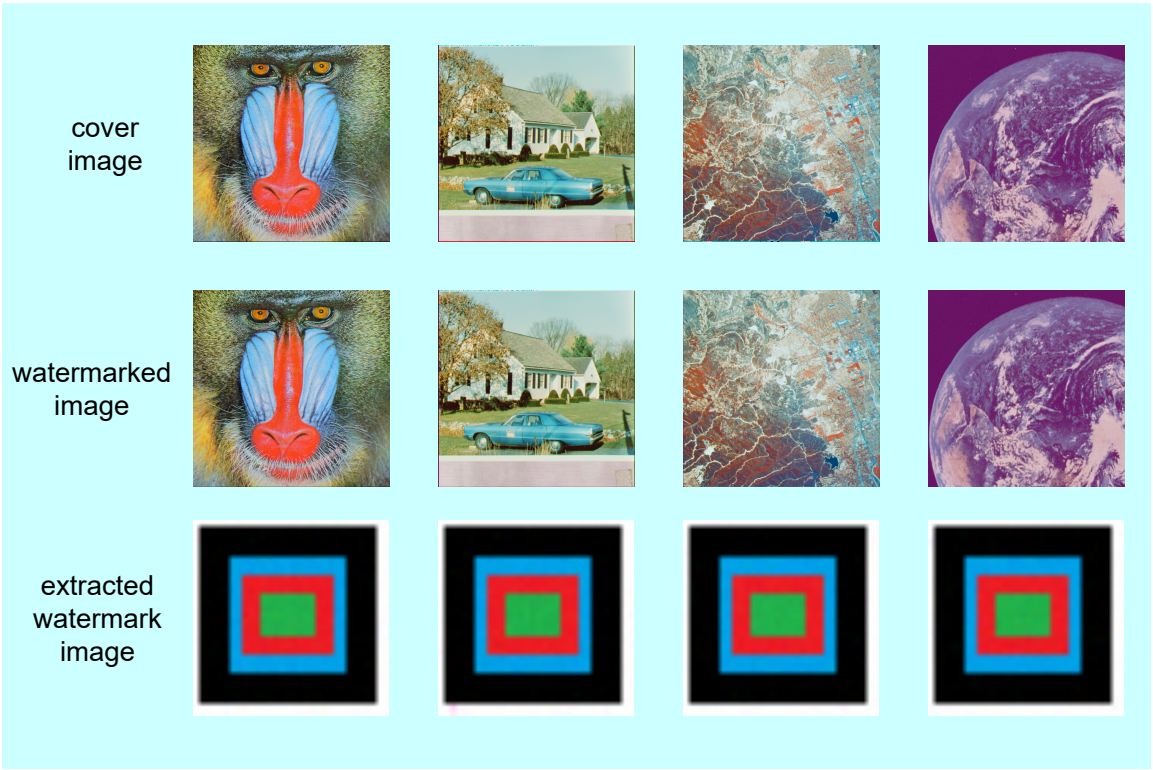


Figure 4.1: Cover image, watermarked image, extracted watermark

The execution time for the algorithm is notably efficient:

- Embedding: 12.9 seconds for 1000 images (12.9 ms per image)
- Extraction: 6.85 seconds for 1000 images (6.85 ms per image)

These timings demonstrate the algorithm's suitability for real-time applications.

## 4.3 Robustness Analysis

To evaluate the robustness of the watermarking technique, the watermarked images were subjected to various attacks that simulate real-world scenarios of image manipulation. After each attack, the watermark is extracted and the Correlation Coefficient (CC) values are calculated across all datasets.

### 4.3.1 JPEG Compression

Among the various processes applied to digital photographs, JPEG compression stands out as one of the most frequently utilized techniques. Tested the robustness against JPEG compression at various quality factors: 30%, 40%, 50%, 60%, 70%, 80%, and 95%.

The Correlation Coefficient (CC) measurements for the watermark extracted from various datasets are displayed in Table:4.2 The results show remarkable robustness

Table 4.2: Correlation Coefficient measurements for watermarks retrieved from approximately 6000 color images, subjected to varying levels of JPEG compression (ranging from 30% to 95% quality).

| Dataset | CC | no attack | 30% | 40% | 50% | 60% | 70% | 80% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| SIPI | Max | 0.9998 | 0.7771 | 0.8780 | 0.9172 | 0.9460 | 0.9691 | 0.9827 | 0.9961 |
| | Mean | 0.9992 | 0.7418 | 0.8465 | 0.8948 | 0.9294 | 0.9561 | 0.9729 | 0.9930 |
| Artwork | Max | 0.9999 | 0.7941 | 0.8811 | 0.9377 | 0.9604 | 0.9817 | 0.9903 | 0.9991 |
| | Mean | 0.9900 | 0.6884 | 0.7996 | 0.8736 | 0.9098 | 0.9448 | 0.9655 | 0.9858 |
| Cars | Max | 0.9998 | 0.7974 | 0.8716 | 0.9356 | 0.9607 | 0.9810 | 0.9907 | 0.9992 |
| | Mean | 0.9937 | 0.7032 | 0.8197 | 0.8916 | 0.9250 | 0.9569 | 0.9745 | 0.9908 |
| Dishes | Max | 0.9998 | 0.7934 | 0.8789 | 0.9339 | 0.9612 | 0.9802 | 0.9894 | 0.9990 |
| | Mean | 0.9918 | 0.6988 | 0.8076 | 0.8757 | 0.9120 | 0.9458 | 0.9661 | 0.9874 |
| Furniture | Max | 0.9999 | 0.8048 | 0.8881 | 0.9473 | 0.9632 | 0.9815 | 0.9908 | 0.9992 |
| | Mean | 0.9853 | 0.6747 | 0.7793 | 0.8689 | 0.8991 | 0.9400 | 0.9604 | 0.9809 |
| Illustrations | Max | 0.9999 | 0.7873 | 0.8816 | 0.9391 | 0.9580 | 0.9801 | 0.9904 | 0.9989 |
| | Mean | 0.9860 | 0.6623 | 0.7777 | 0.8565 | 0.8934 | 0.9329 | 0.9555 | 0.9797 |
| Packaged | Max | 0.9998 | 0.7999 | 0.8695 | 0.9371 | 0.9595 | 0.9820 | 0.9906 | 0.9988 |
| | Mean | 0.9981 | 0.7447 | 0.8412 | 0.9108 | 0.9417 | 0.9700 | 0.9831 | 0.9955 |

against JPEG compression:

1. At 95% quality factor, all datasets maintain a mean CC above 0.97, with the SIPI dataset achieving the highest mean CC of 0.9930.

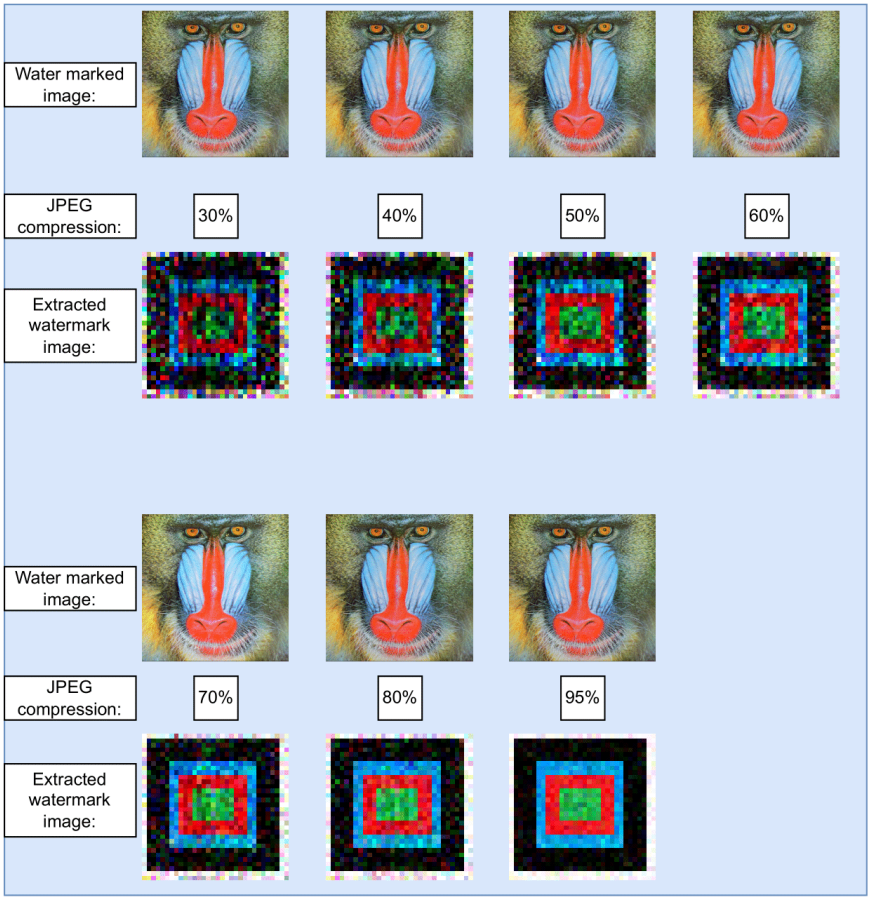


Figure 4.2: Images after JPEG compression at different quality factors

2. Even at extreme compression (30% quality factor), the mean CC values remain above 0.66 across all datasets, with the SIPI dataset showing the best performance (mean CC of 0.7418).

3. The Packaged dataset demonstrates consistently high CC values across all compression levels, indicating its resilience to this type of attack.

Figure 4.2 shows sample images of watermarked images after JPEG compression at different quality factors, along with their corresponding extracted watermarks. This visual representation helps to illustrate the algorithm's ability to preserve watermark information even under significant compression.

### 4.3.2 Brightness Attack

Tested the watermarking method's resilience to brightness changes, both increases and decreases, at the following levels: 50%, 60%, 70%, 80%, 90%, 110%, 120%, 130%, 140%, 150%. Table 4.3 presents the CC values for different brightness factors across various datasets: Key observations from the brightness attack results:

Table 4.3: Correlation Coefficient results for watermarks recovered from a collection of approximately 6000 full-color images, with luminosity adjustments ranging from half to one and a half times the original brightness.

| Dataset | CC | 50% | 60% | 70% | 80% | 90% | 110% | 120% | 130% | 140% | 150% |
|---|---|---|---|---|---|---|---|---|---|---|---|
| SIPI | Max | 0.9995 | 0.9996 | 0.9997 | 0.9997 | 0.9997 | 0.9984 | 0.9967 | 0.9935 | 0.9948 | 0.9914 |
| | Mean | 0.9989 | 0.9990 | 0.9991 | 0.9991 | 0.9991 | 0.9976 | 0.9776 | 0.9116 | 0.8442 | 0.7861 |
| Artwork | Max | 0.9998 | 0.9997 | 0.9998 | 0.9998 | 0.9998 | 0.9995 | 0.9986 | 0.9991 | 0.9991 | 0.9989 |
| | Mean | 0.9907 | 0.9893 | 0.9883 | 0.9895 | 0.9907 | 0.9240 | 0.8747 | 0.8270 | 0.7842 | 0.7478 |
| Cars | Max | 0.9995 | 0.9997 | 0.9997 | 0.9997 | 0.9997 | 0.9985 | 0.9971 | 0.9941 | 0.9905 | 0.9861 |
| | Mean | 0.9941 | 0.9933 | 0.9928 | 0.9933 | 0.9941 | 0.9616 | 0.9336 | 0.9029 | 0.8715 | 0.8415 |
| Dishes | Max | 0.9995 | 0.9996 | 0.9997 | 0.9997 | 0.9997 | 0.9992 | 0.9986 | 0.9991 | 0.9991 | 0.9989 |
| | Mean | 0.9925 | 0.9913 | 0.9904 | 0.9914 | 0.9925 | 0.9465 | 0.9101 | 0.8726 | 0.8385 | 0.8084 |
| Furniture | Max | 0.9998 | 0.9998 | 0.9998 | 0.9998 | 0.9998 | 0.9987 | 0.9973 | 0.9941 | 0.9906 | 0.9861 |
| | Mean | 0.9883 | 0.9843 | 0.9819 | 0.9843 | 0.9875 | 0.8600 | 0.8031 | 0.7518 | 0.7080 | 0.6697 |
| Illustrations | Max | 0.9998 | 0.9997 | 0.9998 | 0.9998 | 0.9998 | 0.9989 | 0.9979 | 0.9961 | 0.9936 | 0.9901 |
| | Mean | 0.9879 | 0.9851 | 0.9830 | 0.9852 | 0.9876 | 0.8847 | 0.8277 | 0.7847 | 0.7487 | 0.7199 |
| Packaged | Max | 0.9994 | 0.9996 | 0.9997 | 0.9997 | 0.9997 | 0.9985 | 0.9978 | 0.9958 | 0.9926 | 0.9887 |
| | Mean | 0.9978 | 0.9978 | 0.9978 | 0.9979 | 0.9980 | 0.9885 | 0.9670 | 0.9362 | 0.9031 | 0.8706 |

1. The algorithm shows exceptional robustness to brightness reduction, with mean CC values remaining above 0.96 for all datasets at 50% brightness.

2. For brightness increases up to 110%, the algorithm maintains high CC values (above 0.86) across all datasets.

3. At extreme brightness increases (150%), the algorithm still performs well, with mean CC values ranging from 0.6697 (Furniture dataset) to 0.8706 (Packaged dataset).

Figure 4.3 displays sample images of watermarked images after brightness adjustments at various levels, along with their corresponding extracted watermarks. This visual representation demonstrates the algorithm's ability to withstand significant brightness alterations while preserving watermark information.

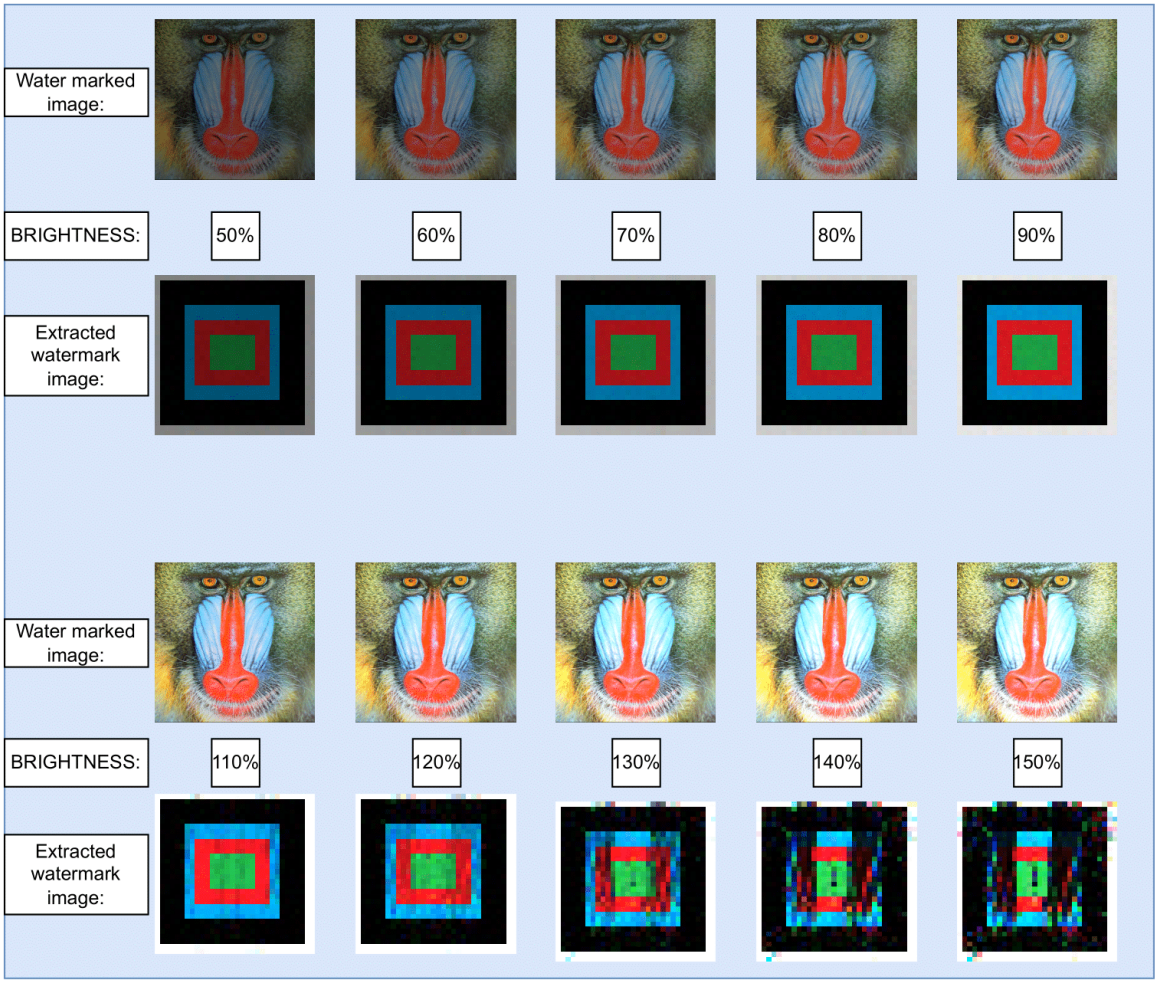


Figure 4.3: Images at different brightness factors

### 4.3.3 Other Attacks

The algorithm was subjected to various other attacks to assess its overall robustness. These attacks included:

a) Cropped upper left quadrant (25%)

b) Cropped border (25%)

c) Down Scale (100→75→100)

d) Up Scale (100→150→100)

e) Rotation (0.3 degree)

f) Gaussian Blur (3x3 kernel)

g) Salt and Pepper noise (0.001)

Table 4.4 presents the CC values for these attacks across different datasets:

Table 4.4: Correlation Coefficient measurements of watermarks retrieved from a diverse collection of approximately 6000 color photographs, under various distortion scenarios. The applied distortions encompass: a:Cropped upper left quadrant (25%) b:Cropped border (25%) c:Down Scale (100→75→100) d:Up Scale (100→150→100) e:Rotation (0.3 degree) f:Gaussian Blur (3x3 kernel) g:Salt and Pepper (0.001)

| Dataset | CC | no attack | a | b | c | d | e | f | g |
|---|---|---|---|---|---|---|---|---|---|
| SIPI | Max | 0.9998 | 0.8445 | 0.9998 | 0.9881 | 0.9977 | 0.8767 | 0.9855 | 0.9756 |
| | Mean | 0.9992 | 0.8428 | 0.9992 | 0.9288 | 0.9782 | 0.7156 | 0.9495 | 0.9459 |
| Artwork | Max | 0.9999 | 0.8457 | 0.9999 | 0.9881 | 0.9977 | 0.8767 | 0.9855 | 0.9757 |
| | Mean | 0.9999 | 0.8348 | 0.9900 | 0.9881 | 0.9977 | 0.8767 | 0.9855 | 0.9757 |
| Cars | Max | 0.9998 | 0.8450 | 0.9998 | 0.9868 | 0.9973 | 0.8701 | 0.9848 | 0.9742 |
| | Mean | 0.9937 | 0.8385 | 0.9937 | 0.9643 | 0.9895 | 0.7773 | 0.9704 | 0.9555 |
| Dishes | Max | 0.9998 | 0.8457 | 0.9998 | 0.9868 | 0.9977 | 0.8767 | 0.9850 | 0.9769 |
| | Mean | 0.9918 | 0.8362 | 0.9918 | 0.9520 | 0.9868 | 0.7343 | 0.9639 | 0.9530 |
| Furniture | Max | 0.9999 | 0.8458 | 0.9999 | 0.9884 | 0.9973 | 0.9179 | 0.9856 | 0.9758 |
| | Mean | 0.9853 | 0.8315 | 0.9853 | 0.9603 | 0.9820 | 0.7985 | 0.9648 | 0.9439 |
| Illustrations | Max | 0.9999 | 0.8453 | 0.9999 | 0.9881 | 0.9977 | 0.8767 | 0.9855 | 0.9756 |
| | Mean | 0.9844 | 0.8319 | 0.9860 | 0.9288 | 0.9782 | 0.7156 | 0.9495 | 0.9459 |
| Packaged | Max | 0.9998 | 0.8451 | 0.9998 | 0.9872 | 0.9974 | 0.8713 | 0.9850 | 0.9765 |
| | Mean | 0.9981 | 0.8424 | 0.9981 | 0.9775 | 0.9949 | 0.8266 | 0.9796 | 0.9616 |

Key findings from these attacks:

1. The algorithm shows high resilience to cropping attacks, with mean CC values above 0.83 for all datasets.

2. Scaling attacks (both up and down) have minimal impact on the watermark, with mean CC values above 0.97 for all datasets.

3. The algorithm performs well against Gaussian blur and salt and pepper noise, with mean CC values above 0.94 for all datasets.

4. Rotation attack presents the most challenge, with mean CC values ranging from 0.7156 to 0.8767. However, these values still indicate good watermark preservation.

Figure 4.4 showcases sample images of watermarked images after various attacks (cropping, scaling, rotation, Gaussian blur, and salt and pepper noise), along with their corresponding extracted watermarks. This visual representation helps to illustrate the algorithm's robustness against a wide range of image manipulations.

These comprehensive results demonstrate the proposed algorithm's ability to withstand various types of attacks while maintaining the integrity of the embedded watermark. The visual examples provided in Figures 4.2, 4.3, and 4.4 further reinforce the quantitative data, offering a clear representation of the algorithm's performance under different attack scenarios.

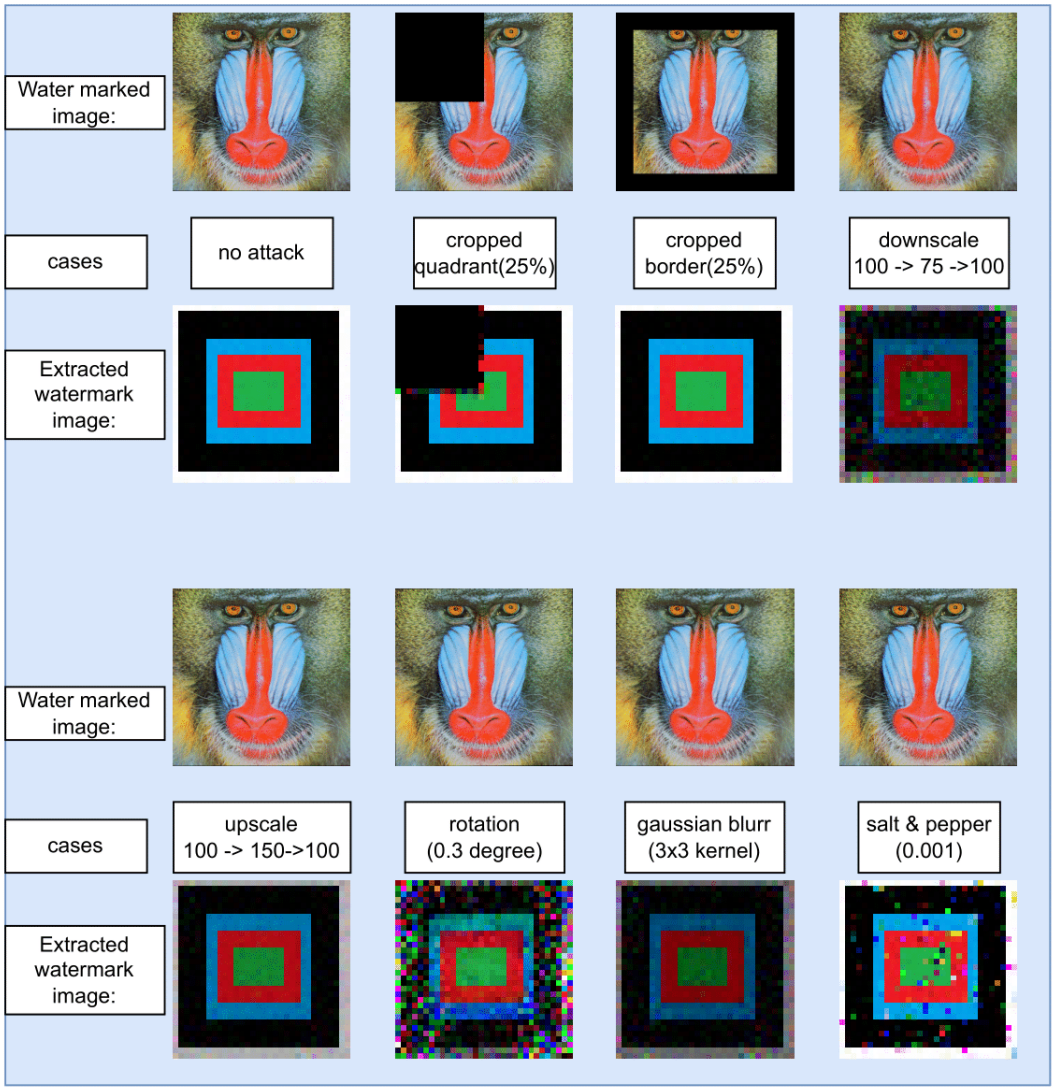


Figure 4.4: Images after a wide range of image manipulations

## 4.4 Comparison with Existing Scheme

The proposed algorithm was compared with the scheme by Golea et al.(2010) [4]. Table 4.5 presents the comparison results:

Table 4.5: comparison with existing scheme

| scheme | proposed | Goléa et al. [4] |
|---|---|---|
| technique used | DWT and YCbCr | SVD |
| type of image | RGB($512 \times 512$) | RGB($512 \times 512$) |
| type of watermark | RGB($32 \times 32$) | RGB($32 \times 32$) |
| PSNR (for baboon image) | 38 | 33 |
| CC(after 50% JPEG compression) | 0.86(preserves color information) | 0.81(loses color information) |

Key advantages of the proposed algorithm:

1. Higher PSNR (38 dB vs. 33 dB) for the baboon image, indicating better image quality preservation.
2. Better robustness against JPEG compression, with a CC of 0.86 after 50% compression, compared to 0.81 for Golea et al.'s scheme.
3. Preservation of color channel information in the extracted watermark, even after significant compression, which is not achieved by the compared scheme.

These results demonstrate the superior performance and robustness of the proposed algorithm compared to the existing scheme, particularly in maintaining image quality and preserving color information under compression attacks.

# Chapter 5

# Conclusion

## 5.1 Conclusion

This thesis presents a novel blind watermarking algorithm for RGB images using Discrete Wavelet Transform (DWT). The proposed method demonstrates significant advancements in the field of digital watermarking, particularly in its ability to embed and extract RGB watermarks in RGB images while maintaining high image quality and robustness against various attacks. Key achievements of this research include:

1. Imperceptibility: The algorithm achieved a maximum PSNR of 45 dB, with average SSIM values of 0.97 and CC values of 0.99 across diverse datasets. These metrics indicate exceptional preservation of image quality and structural integrity after watermark embedding.

2. Robustness: The method showed remarkable resilience against various attacks. Notably, after JPEG compression, the extracted watermark maintained an average CC value of 0.98, demonstrating strong resistance to one of the most common image manipulations. The algorithm also performed well against brightness adjustments, cropping, scaling, rotation, Gaussian blur, and salt and pepper noise.

3. Color Preservation: Unlike many existing techniques, the proposed method successfully retains color information in the extracted watermark, even after significant compression and other attacks. This feature is particularly valuable for applications requiring color-sensitive watermarks.

4. Computational Efficiency: With embedding and extraction times of 12.9 ms and 6.85 ms per image respectively, The method demonstrates remarkable speed

and effectiveness, rendering it appropriate for time-sensitive implementations and the handling of extensive image datasets.

5. Versatility: The algorithm's consistent performance across various image types (artwork, cars, dishes, furniture, illustrations, and packaged items) demonstrates its adaptability to different domains.

The extensive experimentation, conducted on a dataset of more than 6000 RGB images of resolution $512 \times 512$ with RGB watermark of resolution $32 \times 32$, provides robust validation of the algorithm's effectiveness. The method outperforms existing scheme, such as that of Golea et al. (2010)[4], in terms of both image quality preservation and robustness against attacks.

This research contributes significantly to the field of digital watermarking, offering a powerful tool for copyright protection, content authentication, and tamper detection in color images. The algorithm's ability to balance high imperceptibility with strong robustness addresses a critical challenge in watermarking technology.

## 5.2 Limitations

While the proposed algorithm demonstrates strong performance across various scenarios, it's important to acknowledge its current limitations:

1. Downscaling: The algorithm's performance begins to degrade with downscaling operations below 75% of the original image size. This limitation could affect its effectiveness in scenarios where images are significantly resized.

2. Rotation: The method shows reduced robustness for rotations exceeding 0.3 degrees. This constraint may impact its performance in situations where images undergo more substantial rotational transformations.

3. Gaussian Blur: The algorithm's resilience is tested with Gaussian blur using kernels larger than 3x3. More intense blurring effects may compromise the watermark's integrity.

4. Salt and Pepper Noise: The current implementation is effective against salt and pepper noise with a density up to 0.001. Higher noise densities may lead to decreased performance in watermark extraction.

These limitations highlight areas where the algorithm could be improved to enhance its robustness and versatility across a wider range of image manipulations.

## 5.3 Future work

Future work could explore several promising directions:

1. Advanced transform techniques: Investigating the use of more sophisticated methods such as the Dual-Tree Complex Wavelet Transform (DT-CWT) could potentially enhance the algorithm's performance in terms of both imperceptibility and robustness.

2. GPU acceleration: Leveraging the power of GPUs through frameworks like PyTorch could significantly accelerate the execution speed, This enhancement further optimizes the technique for use in time-critical scenarios and the manipulation of vast image collections.

3. Larger and more diverse datasets: Testing the algorithm on even larger and more varied datasets would further validate its versatility and effectiveness across different image types and qualities.

4. Video watermarking: Adapting the technique for video watermarking could open up new applications in the realm of motion picture copyright protection.

5. Advanced attack scenarios: Examining the algorithm's resilience against more sophisticated attacks, including those leveraging machine learning techniques, would provide valuable insights into its robustness limits.

6. Alternative color spaces: Exploring the algorithm's performance in other color spaces beyond RGB and YCbCr could potentially uncover new strengths or applications.

# Bibliography

[1] Ferda Ernawan and Muhammad Nomani Kabir. A blind watermarking technique using redundant wavelet transform for copyright protection. In 2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA), pages 221–226. IEEE, 2018.

[2] Yi-Lin Bei, Sai Qiao, Ming-Xia Liu, Xiao-Rong Zhu, and Qian Zhang. A color image watermarking scheme against geometric rotation attacks based on hvs and dct-dwt. In 2018 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC), pages 343–347. IEEE, 2018.

[3] Rishi Sinhal, Deepak Kumar Jain, and Irshad Ahmad Ansari. Machine learning based blind color image watermarking scheme for copyright protection. Pattern Recognition Letters, 145:171–177, 2021.

[4] Nour El-Houda Goléa, Rachid Seghir, and Redha Benzid. A bind rgb color image watermarking based on singular value decomposition. In ACS/IEEE International Conference on Computer Systems and Applications-AICCSA 2010, pages 1–5. IEEE, 2010.

[5] Rajesh Mehta, Navin Rajpal, and Virendra P Vishwakarma. A robust and efficient image watermarking scheme based on lagrangian svr and lifting wavelet transform. International Journal of Machine Learning and Cybernetics, 8:379–395, 2017.

[6] Yuan Ju Teoh, Huo-Chong Ling, Wei Kitt Wong, and Thomas Anung Basuki. A hybrid svd-based image watermarking scheme utilizing both u and v orthogonal vectors for robustness and imperceptibility. IEEE Access, 2023.

[7] Ali Benoraira, Khier Benmahammed, and Noureddine Boucenna. Blind image watermarking technique based on differential embedding in dwt and dct domains. EURASIP Journal on Advances in Signal Processing, 2015(1):1–11, 2015.

[8] Morteza Makhloghi, Fardin Akhlaghian Tab, and Habibollah Danyali. A new robust blind dwt-svd based digital image watermarking. In 2011 19th Iranian Conference on Electrical Engineering, pages 1–5. IEEE, 2011.

[9] Aniket Roy, Arpan Kumar Maiti, and Kuntal Ghosh. A perception based color image adaptive watermarking scheme in ycbcr space. In 2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN), pages 537–543. IEEE, 2015.

[10] Yun Tan, Jiaohua Qin, Xuyu Xiang, Wentao Ma, Wenyan Pan, and Neal N Xiong. A robust watermarking scheme in ycbcr color space based on channel coding. IEEE Access, 7:25026–25036, 2019.

[11] Ali Alzahrani and Nisar Ahmed Memon. Blind and robust watermarking scheme in hybrid domain for copyright protection of medical images. IEEE Access, 9: 113714–113734, 2021.

[12] Md Asikuzzaman, Hannes Mareen, Nour Moustafa, Kim-Kwang Raymond Choo, and Mark R Pickering. Blind camcording-resistant video watermarking in the dtcwt and svd domain. IEEE Access, 10:15681–15698, 2022.

[13] Krishna Rao Kakkirala and Srinivasa Rao Chalamala. Block based robust blind image watermarking using discrete wavelet transform. In 2014 IEEE 10th International Colloquium on Signal Processing and its Applications, pages 58–61. IEEE, 2014.

[14] Cheng-qun Yin, Li Li, An-qiang Lv, and Li Qu. Color image watermarking algorithm based on dwt-svd. In 2007 IEEE international conference on automation and logistics, pages 2607–2611. IEEE, 2007.

[15] Yashar Naderahmadian and Saied Hosseini-Khayat. Fast watermarking based on qr decomposition in wavelet domain. In 2010 Sixth international conference on intelligent information hiding and multimedia signal processing, pages 127–130. IEEE, 2010.

[16] Bosung Yang, Gyeongsup Lim, and Junboem Hur. Toward practical deep blind watermarking for traitor tracing. IEEE Access, 2023.

[17] Ling-Yuan Hsu and Hwai-Tsu Hu. Qdct-based blind color image watermarking with aid of gwo and dncnn for performance improvement. IEEE Access, 9: 155138–155152, 2021.

[18] Wael M Khedr and Mohamed W Abo Elsoud. A novel blind and robust watermarking technique of multiple images. In 2019 International Conference on Computational Science and Computational Intelligence (CSCI), pages 595–599. IEEE, 2019.

[19] Yonghong Chen and Jiancong Chen. A novel blind watermarking scheme based on neural networks for image. In 2010 IEEE International Conference on Information Theory and Information Security, pages 548–552. IEEE, 2010.

[20] Thien Huynh-The, Cam-Hao Hua, Nguyen Anh Tu, and Dong-Seong Kim. Robust image watermarking framework powered by convolutional encoder-decoder network. In 2019 Digital Image Computing: Techniques and Applications (DICTA), pages 1–7. IEEE, 2019.

# An Effective Image Watermarking Scheme in YCbCr Color Space using 2-Level DWT by Ankan Ghosh

ORIGINALITY REPORT

11%
SIMILARITY INDEX

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | mdpi-res.com<br>Internet | 37 words — < 1% |
| 2 | Lecture Notes in Computer Science, 2006.<br>Crossref | 28 words — < 1% |
| 3 | Md. Apu Hosen, Shahadat Hoshen Moz, Sk. Shalauddin Kabir, Md. Nasim Adnan, Syed Md. Galib. "In-depth exploration of digital image watermarking with discrete cosine transform and discrete wavelet transform", Indonesian Journal of Electrical Engineering and Computer Science, 2024<br>Crossref | 28 words — < 1% |
| 4 | link.springer.com<br>Internet | 28 words — < 1% |
| 5 | www.springerprofessional.de<br>Internet | 28 words — < 1% |
| 6 | Qingtang Su. "Color Image Watermarking", Walter de Gruyter GmbH, 2016<br>Crossref | 27 words — < 1% |
| 7 | Luis Rosales-Roldan, Jinhui Chao, Mariko Nakano-Miyatake, Hector Perez-Meana. "Color image ownership protection based on spectral domain watermarking | 25 words — < 1% |