Collusion Resistant Watermarking Using Convolutional Encoding and Random Spreading

Abdul Rehman, Gaëtan Le Guelvouit, Jean Dion, Frédéric Guilloud, Matthieu Arzel

Collusion Resistant Watermarking Using Convolutional Encoding and Random Spreading

Abdul Rehman
IRT b-com
Cesson-Sevigné, France
abdul.rehman@b-com.com

Gaëtan Le Guelvouit
IRT b-com
Cesson-Sevigné, France
gaetan.leguelvouit@b-com.com

Jean Dion
IRT b-com
Cesson-Sevigné, France
jean.dion@b-com.com

Frédéric Guilloud
IMT Atlantique, Lab-STICC
Brest, France
frederic.guilloud@imt-atlantique.fr

Matthieu Arzel
IMT Atlantique, Lab-STICC
Brest, France
matthieu.arzel@imt-atlantique.fr

***Abstract*—This paper presents a Discrete Wavelet Transform based collusion resistant video watermarking to trace colluders involved in unauthorized video distribution. Our scheme uses Tardos-Skoric codes as fingerprints. To reduce the errors on the fingerprinting codes, we propose a joint scheme that combines pseudo-random spreading sequences and convolutional codes. The performance when the fingerprint embedding and the attack are simulated as a binary symmetric channel proves that the proposed scheme performs better in terms of bit error rate and in terms of colluders tracing using binary attacks. Simulations of a darken attack on the watermarked videos show promising results for low to moderate opacities of the fingerprint embedding.**

Keywords*—*Video watermarking; collusion; fingerprinting codes; convolutional encoding; spreading.

I. Introduction

In the age of wide digital content distribution, it is now more crucial than ever to provide reliable and powerful techniques to prevent unauthorized redistribution of multimedia objects [1] [2] denoted as collusion attacks. In collusion attacks, multiple users merge their content to alter the watermark, making it difficult to trace these users who are the source of unauthorized copies and thus posing a significant threat to traditional watermarking methods [3] [4]. Tardos codes, proposed in 2003 by Tardos [5], are collusion-resistant codes which were the first theoretically proven codes to efficiently prevent illegal redistribution of digital content. The principle of collusion-resistant watermarking is to associate a unique fingerprint per subscriber into each copy of the content. After the collusion attack, Tardos codes enable the content distributor to retrieve the subscribers responsible for the creation and redistribution of the illegal content. The length of Tardos codes is given by [5] as $100{c_0}^2 \ln \frac{1}{\varepsilon_1}$ and depends on the number of colluders to trace, c_0, and on the probability of accusing any innocent user, ε_1. Tracing more colluders implies higher fingerprinting lengths and brings more difficulties to hide the fingerprint into the video. Later on, Skoric et al. [6] reduced the Tardos code length approximatelt 5 times taking into account the number of users n, coming up with a length $\frac{1}{2}\pi^2{c_0}^2 \ln \frac{n}{\varepsilon_1}$. In this study, we use Skoric codes as fingerprinting codes against collusion attacks.

The problem is that embedding fingerprints into videos adds noise yielding binary errors on the fingerprint, and consequently decreasing the performance of Tardos-Skoric codes. This is the reason why numerous studies take advantage of pseudo-random spreading [7]–[11] to hide data into images with low errors after retrieving the data. In a recent study [14], we also used random spreading to hide Tardos-Skoric generated fingerprints in a watermark image and found out the best generator-decoder combination of collusion codes for real time implementation to find at-least one colluder. However, the gain provided by random spreading is decreased as the fingerprint length is increased for a given image size. To improve the performance of random spreading, we propose to use Error Correcting Codes (ECC).

The authors in [12] demonstrated that using convolutional encoding with Discrete Wavelet Transform (DWT) watermarking provided enhanced resistance to multimedia compression, but without addressing other crucial attacks, such as collusion, geometric distortions and cropping. In [13], the authors illustrate the robustness of a watermarking scheme for images using convolutional codes embedding, and considering all standard multimedia attacks. However, collusion attacks are not adressed. Also, watermarking in this study is non-blind, meaning that the original image is required. In this article, we focus on blind watermarking: the original content is not available at the receiver side. Convolutional encoding and random spreading are also combined in [15] to lower the bit error rate brought on by interference from the host signal. However, in the simulations, no attacks of any kind were taken into account. In this work, we propose to use convolutional codes either concatenated with random spreading or jointly, as proposed in [15], to improve colluders tracing in collusion attacks.

Our paper is organized as follows. In Section II, we review the fundamentals of watermarking and illustrate the need for spreading when using collusion codes. Then, we describe the two proposed spreading schemes using convolutional codes: a joint scheme and a concatenated spreading schemes. The performance of these spreading schemes is presented in Sec-

 ISBN: 978-1-68558-131-2

tion III, first over a binary symmetric channel, and then for collusion tracing in a realistic scenario where the fingerprint is embedded into the videos and where a real collusion attack is performed. Conclusions and perspectives are presented in Section IV.

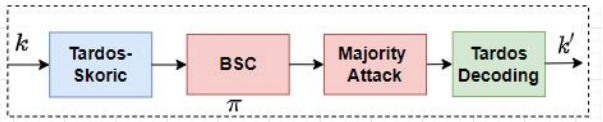
II. Combining Coding and Spreading for Watermarking

A. Motivation

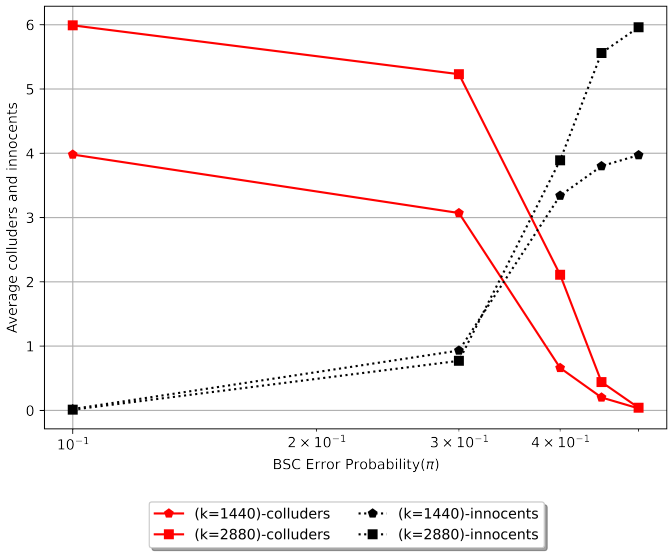
1) Definition and Notations: In collusion resistant video watermarking, a Tardos-Skoric fingerprinting code of length k is hidden into an image to trace the colluders. To trace a higher number of colluders among many users requires higher k. The maximum value of k to be hidden in a $360p$ image corresponds to the image size given by 360×640. The watermarked image I_{wt} is obtained by alpha blending $I_{wt} = \mathscr{O} I_i + (1-\mathscr{O}) I_w$, where $\mathscr{O}$ is the opacity ranging between 0 and 1, I_w is the watermark image (the information to be hidden, e.g., the fingerprint code) and I_i is the original image. We employed DWT with level-3 to obtain I_w from the fingerprint. Note that we omit the LL3 (lowest frequency) part of the DWT since it cannot be reproduced in a blind detection of the watermark image [16]. We thus come up with the final watermark image size $m = (360 \times 640) - LL3 = 226800$ corresponding to the maximum value for the fingerprint code k. However, watermarking should be discrete. To accomplish this, the opacity $\mathscr{O}$ has to be very close to 1 yielding a power ratio between the watermark image I_w and the watermarked image I_{wt} to be as low as about -20dB, depending on the considered image.

2) Impact of binary errors on the Tardos-Skoric codes: For such low SNRs, many errors occur in the fingerprint, which leads to dramatic performance for collusion tracing with Tardos-Skoric codes. We analyzed the performance of Tardos-Skoric codes using the majority vote attack to trace out the colluders without any spreading. The simulation model is depicted in Figure 1(a): Tardos-Skoric codes are modified by a Binary Symetric Channel (BSC) with error probability π, which represents the possible errors due to the embedding with a low Signal over Noise Ratio (SNR). Figure 1(b) illustrates that, whatever the length of the fingerprint k, the number of average detected colluders drops for binary error probabilities higher than $\pi = 2 \times 10^{-1}$, which corresponds to a much higher SNR than -20dB.

A well known way to improve SNR is to spread the fingerprint length k over the image length m. Let α denote the spreading rate $\alpha = \frac{k}{m}$. A lower α results in fewer binary errors on the watermark image but, but as the image length is fixed, it also resultsin lower fingerprint k, reducing total colluders detecting capacity. In this Section, firstly, we propose to improve the efficiency of the spreading by combining pseudo-random sequences with ECC. Then, we address the issue of optimizing the spreading rate α for a fixed image length m.



(a) Simulation model



(b) Simulation results

Figure 1: Colluders tracing without spreading scheme for majority vote attack: (a) Simulation model (b) Average detected colluders with $k = [1440, 2880], n = 1000$ and $\varepsilon_1 = 10^{-3}$.

B. Proposed Spreading Schemes

1) Convolutional Codes and Viterbi Decoding: Although convolutional codes have been surpassed by many others, they are still often used in watermarking. The authors from [17]–[19] showed that using convolutional codes with fragile watermarking improves the SNR but also that the scheme is not robust to compression, contrast enhancement and collusion attacks. In this paper, we propose to use convolutional codes to increase the robustness against noise on embedded collusion codes. A convolutional code [20] is specified by its coding rate r_{cc} and the depth N of its shift register. The trellis diagram is a result of expanding the convolutional code state diagram in time. The number of the states in the trellis diagram is 2^N. To decode a convolutional code, we generally use the Viterbi algorithm [20], which finds efficiently the shortest path on the trellis diagram.

2) Combination of coding and pseudo-random sequences: Two spreading schemes using convolutional codes are proposed hereafter: the concatenated scheme and the joint scheme.

For the concatenated scheme, the convolutional encoder is utilized to encode the k bits of the fingerprint with rate r_{cc}. The trellis shown in Figure 2(a) illustrates the outputs of the convolutional encoder of rate $r_{cc} = 1/2$ with $N = 2$ for each possible transition between 2 states. The output of the encoder is then spread using pseudo-random sequences of rate $\frac{\alpha}{r_{cc}}$.

In the joint scheme, the k bits of fingerprint are encoded and

spread simultaneously utilizing joint convolutional encoding and spreading with rate α, as proposed by [15]. In this scheme, the outputs are given pseudo-random sequences s_{r_i} with $i \in [1,\ldots,2\times 2^N]$ of rate $\frac{1}{\alpha}$, as illustrated in Figure 2(b).

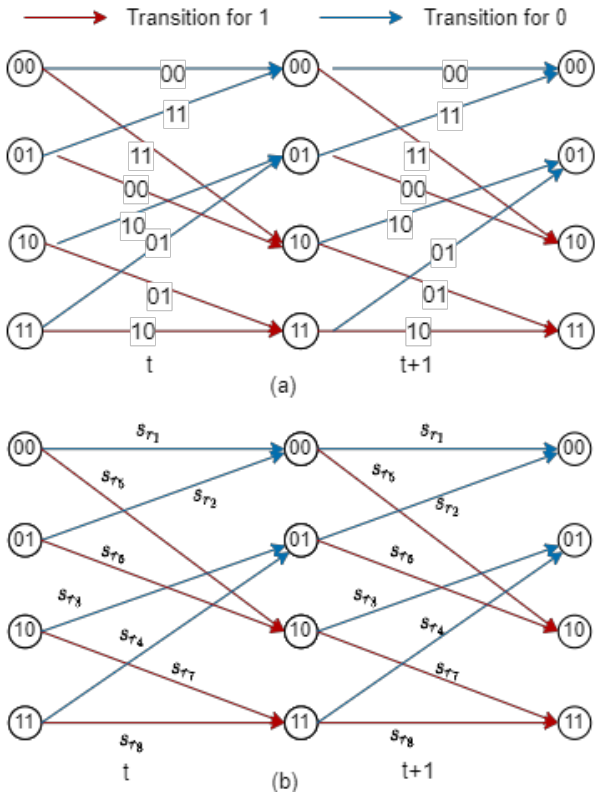


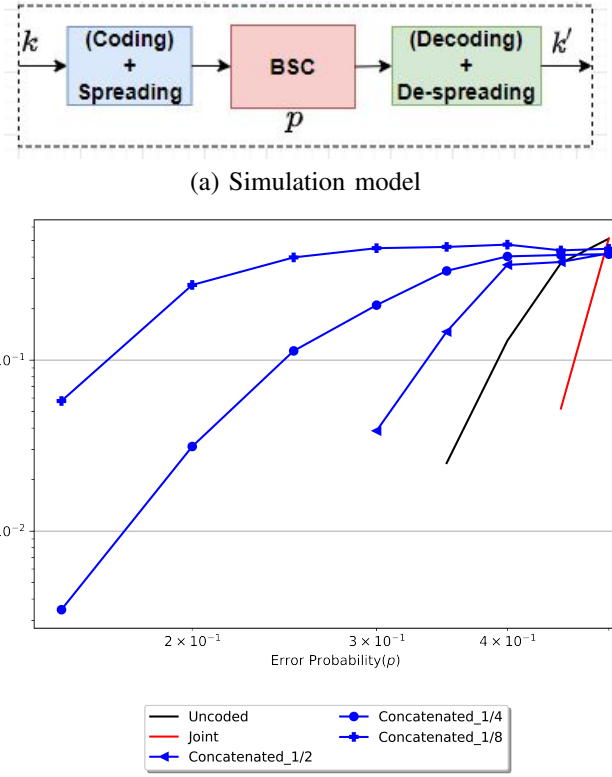
Figure 2: Trellis diagram for 4 states: (a) concatenated scheme with rate $r_{cc} = \frac{1}{2}$ (b) joint scheme with rate $\frac{1}{\alpha}$.

C. Performance comparison

We evaluated the performance of the spreading using a Binary Symmetric Channel (BSC) with error probability p, as shown in Figure 3(a). For performance evaluation, the concatenated and joint schemes are compared with a pure pseudo-random spreading scheme denoted "uncoded" since no convolution code is used.

For the concatenated scheme, we compare three different numbers of shift register: $N = 3,5$ and 9, respectively, associated with 3 different rates $r_{cc} = \frac{1}{2}, \frac{1}{4}$ and $\frac{1}{8}$. The random spreading rates after encoding are thus, respectively, $2\alpha, 4\alpha$ and 8α. Simulations have been performed for $\alpha = 1/157$ (k being set to 1440) and are illustrated in Figure 3(b). The joint scheme clearly outperforms the two other schemes. Note, however, that the uncoded scheme outperforms the concatenated scheme, thus showing that convolutional codes not always improve the spreading performance.

To determine a viable fingerprint length with an acceptable BER, we investigated the trade-off between spreading rate and BER. We performed simulations considering different fingerprint lengths $k = [256, 512, 1024, 1440, 2880]$. Considering the BER target of 2.10^{-1}, the selected possible fingerprint lengths



(a) Simulation model

(b) Simulation results

Figure 3: Bit Error Rate (BER) for spreading schemes combined with convolutional codes and compared to a pure random spreading scheme (uncoded) BSC with error probabilities p for a spreading rate $alpha = 1/157$.

are $k = 2880$ for the joint scheme and $k = 1440$ for the uncoded scheme, as shown in Figure 4. These trade-offs correspond to a spreading rate of $alpha = [1/157, 4/315]$, respectively, for lengths $k = [1440, 2880]$.

III. Performance of Spreading Schemes for Colluders Tracing

Taking the two fingerprint lengths of Tardos-Skoric fingerprints with $n = 1000$ users and a probability of accusing innocent users set to $\varepsilon_1 = 10^{-3}$, we can trace a maximum of $c_0 = [4,6]$ colluders [6]. In this section, we address colluder tracing performance by first emulating video embedding and attacks thanks to a BSC, and then by simulating the realistic embedding (alpha blending and darken attack). To trace the colluders, we used the Nearest Neighbor Search (NNS) decoder for its higher tracing rate and lower complexity [21].

To analyse the impact of spreading scheme for colluders tracing, the fingerprint k is spread and noised over a BSC with error probability p before a majority vote collusion attack is performed, as illustrated in Figure 5(a).

The performance over the BSC is illustrated in the upper graph of Figure 6: we observe that colluders tracing is much improved by the proposed joint scheme compared to the state

 ISBN: 978-1-68558-131-2

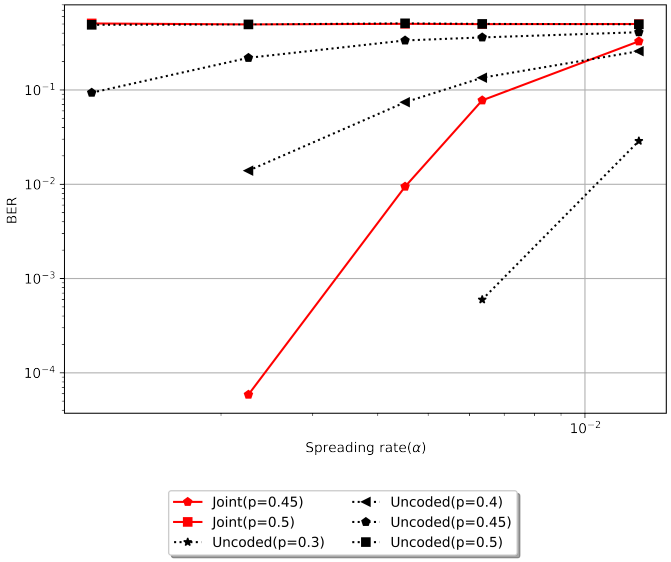
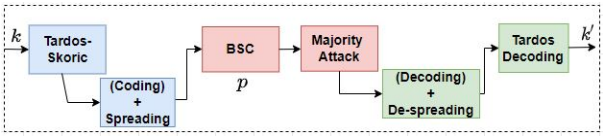
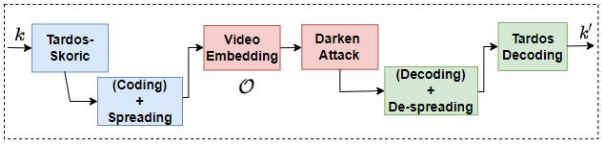


Figure 4: Trade-off between BER and spreading rate for the joint and uncoded scheme with error probability $p \in [0.05, ...0.5]$ for BSC.



(a) Majority attack



(b) Darken attack

Figure 5: Simulation model for colluders tracing with $k = [1440, 2880], n = 1000$ and $\varepsilon_1 = 10^{-3}$ for the uncoded and the joint coding and spreading schemes: (a) Majority vote attack over BSC (b) Darken attack on video with FFMpeg and alpha blending embedding.

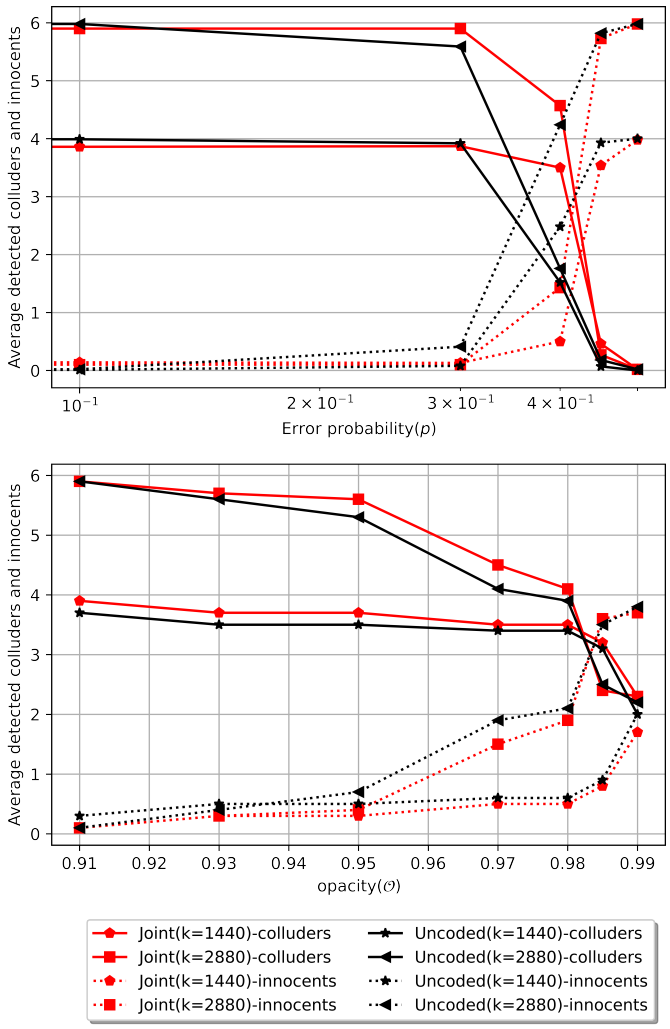


Figure 6: Results for the two simulation models of colluders tracing: The upper part is for the majority vote over BSC, while the plot below is for darken attack on the video with FFMpeg.

of the art uncoded spreading scheme, even when binary errors are higher than $\pi = 2 \times 10^{-1}$.

In the realistic setup, the watermark image is embedded into an open source $1080p$ video Tear of Steel [22]. The watermarked video is created using blending filter of the FFMpeg with alpha channel as opacity. A darken attack using FFMpeg is performed to create an illegal copy of a video as explained in [Mode:B and Table V in [14]]. The model is illustrated in Figure 5(b). In the simulation, we also consider two fingerprint lengths and we let the opacity range from 0.90 to 0.99. The simulation results are depicted in the lower graph of Figure 6. As the opacity increases close to 1 (thus decreasing the SNR), the performance between the 2 schemes becomes equivalent for both fingerprint lengths. However, for lower opacities (higher SNR), the joint scheme still outperforms the uncoded spreading scheme.

IV. Conclusion and Future Work

Tardos-Skoric codes are used to identify the colluders who took part in the collusion to unlawfully redistribute pirated copies of multimedia contents like e.g., videos. Discreetly watermarking the videos with these codes implies a very low SNR. Spreading schemes on Tardos-Skoric codes improve the SNR at the cost of reducing the Tardos-Skoric code length and the tracing performance. In this article, we proposed to combine error correcting codes with pseudo-random spreading to improve the colluder tracing performance. Firstly, we analyzed the trade-off between the spreading rate and the bit error rate on the fingerprint code. We then estimated the performance of the proposed joint convolutional code and random spreading compared to the uncoded random spreading scheme. Performances were obtained first on a 360p image over a binary symmetric channel with a majority vote attack, and then on a 1080p video with Discrete Wavelet Transform

 ISBN: 978-1-68558-131-2

embedded using alpha blending with a darken attack. The performance results in terms of colluder tracing showed that the proposed joint scheme outperforms the uncoded one. Perspectives to this work include the use of more powerful error correction coding schemes to be jointly combined with random spreading.

REFERENCES

[1] Y. Uchida, K. Takagi, and S. Sakazawa, "Fast and accurate content-based video copy detection using bag-of-global visual features," in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2012, pp. 1029–1032.

[2] I. Sayahi, A. Elkefi, and C. Ben Amar, "Join cryptography and digital watermarking for 3D multiresolution meshes security". In: 19th International Conference on Image Analysis and Processing, ICIAP 2017. Springer Verlag. 2017, pp. 637–647.

[3] P. Bas, T. Furon, and F. Cayre, "Practical key length of watermarking systems," in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2012, pp. 1769-1772.

[4] F. Chaabane, M. Charfeddine, and C. Ben Amar, "The impact of error correcting coding in audio watermarking". In: 2011 3rd International Conference on Next Generation Networks and Services (NGNS). 2011, pp. 90z-95, DOI:10.1109/NGNS.2011.6142556.

[5] G. Tardos, "Optimal probabilistic fingerprint codes," in Symposium on the Theory of Computing, 2003, pp 1–24, DOI:10.1145/1346330.1346335.

[6] B. Skoric, S. Katzenbeisser, and M. U. Celik, "Symmetric tardos fingerprinting codes for arbitrary alphabet sizes," Cryptology ePrint Archive, Paper 2007/041, 2007.

[7] A. Gutub, "Boosting image watermarking authenticity spreading secrecy from counting-based secretsharing," CAAI Transactions on Intelligence Technology, vol. 8, pp. 440–452, 2022.

[8] A. Kuznetsov et al., "Adaptive pseudo-random sequence generation for spread spectrum image stenography," in IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), 2020, pp. 161–165.

[9] Y. Huang, B. Niu, H. Guan, and S. Zhang, "Enhancing image watermarking with adaptive embedding parameter and psnr guarantee," IEEE Transactions on Multimedia, vol. 21, no. 10, pp. 2447—2460, 2019.

[10] S. Panchikkil, V.M. Manikandan, and Y-D. Zhang, "A pseudo-random pixel mapping with weighted mesh graph approach for reversible data hiding in encrypted image, Multimedia Tools and Applications, vol. 81, no. 12, pp. 16279–16307, 2022.

[11] R. Venkatesan and M.H. Jakubowski, "Randomized detection for spread-spectrum watermarking: defending against sensitivity and other attacks [image watermarking applications]," in Proceedings. (ICASSP 05). IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 2, pp. ii/9-–ii12 , 2005.

[12] H. Tribak, Y. Zaz, and H. Kelkoul, "Advanced video watermarking approach based on convolutional encoding : Search for new solution against cinematography piracy traffic," in 2018 6th International Conference on Multimedia Computing and Systems (ICMCS), pp. 1–7, 2018.

[13] Y. Tan et al., "A robust watermarking scheme in ycbcr color space based on channel coding," IEEE Access, vol. 7, pp. 25026–25036, 2019.

[14] A. Rehman, G. Le Guelvouit, J. Dion, F. Guilloud, and M. Arzel, "Dwt collusion resistant video watermarking using tardos family codes," in 2022 IEEE 5th International Conference on Image Processing Applications and Systems (IPAS), vol. 5, pp. 1–6, 2022.

[15] G. Le Guelvouit and S. Pateux, "Wide spread spectrum watermarking with side information and interference cancellation," in Security and Watermarking of Multimedia Contents V. SPIE, 2003, vol. 5020, pp. 278–289.

[16] R. Kumar and J. Yadav, "Effective compression and decompression coding techniques using multilevel dwt decomposition and dct," International Journal of Signal and Imaging Systems Engineering, vol. 12, no.3, pp. 71-80, 2021.

[17] K. Yu, L. Chen, Z. Fu, Y. Wang, and T. Lu, "A coding layer robust reversible watermarking algorithm for digital image in multi-antenna system," Signal Processing, vol. 199, pp. 108630, 2022.

[18] Md. Ahasan Kabir, "An efficient low bit rate image watermarking and tamper detection for image authentication," SN Applied Sciences, vol. 3, no. 4, pp. 400, 2021.

[19] J.R. Hernandez, J.F. Delaigle, and B. Macq, "Improving data hiding by using convolutional codes and soft-decision decoding". In: Security and Watermarking of Multimedia Contents II. vol. 3971, SPIE. 2000, pp. 24–47.

[20] A. Bensky, "Chapter 9 - introduction to information theory and coding," in Short-range Wireless Communication, Alan Bensky, Ed., pp. 211–236. Newnes, third edition, 2019.

[21] T. Laarhoven, "Nearest neighbor decoding for tardos fingerprinting codes," Proceedings of the ACM workshop on information hiding and multimedia security, pp. 182-187, 2019.

[22] Blender Foundation, "Tears of steel: Open movie free to share and show 2012," in https://mango.blender.org/download/, retrieved: Feb, 2024.

 ISBN: 978-1-68558-131-2