Fake Document Detection Using SVD and Live Scene Entropy

Samyugtha J
Department of Electrical and Electronics Engineering
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham, India
cb.en.u4eee23154@cb.students.amrita.edu

Vidhyarth S.E
Department of Electrical and Electronics Engineering
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham, India
cb.en.u4elc23057@cb.students.amrita.edu

Deepana S
Department of Electrical and Electronics Engineering
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham, India
cb.en.u4eee23109@cb.students.amrita.edu

Nivetha R
Department of Electrical and Electronics Engineering
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham, India
cb.en.u4eee23122@cb.students.amrita.edu

***Abstract*—The increased use of digital documents in academic, financial, and governmental sectors has raised the risk of forgery and unauthorized copying. Current methods, like QR codes or static watermarks, do not provide true protection against tampering once a document is duplicated or altered. This paper presents a new way to detect fake documents using Singular Value Decomposition (SVD) and live scene entropy. The proposed system embeds a unique entropy signature, taken from a real-time scene, into the document's SVD domain. This entropy-based feature serves as a one-time dynamic watermark that connects the document to its creation context. Using SVD provides strong mathematical reliability, while entropy adds unpredictability, which improves the system's resistance to forgery and copying. Experimental tests show that this method maintains image quality, lowers computational demands, and offers high assurance of authenticity for important documents like certificates and identification records.**

***Index Terms*—Fake document detection, SVD, entropy, digital watermarking, authentication, document security**

I. Introduction

The authenticity of digital documents has become a major concern in today's data-driven world. Certificates, identity proofs, and legal documents are often transmitted electronically, increasing exposure to forgery and manipulation. Traditional verification techniques rely on static identifiers such as signatures, QR codes, or embedded patterns, which can easily be replicated once their structure is known. As a result, the development of a robust, efficient, and context-aware document authentication system is essential.

Singular Value Decomposition (SVD) has emerged as a powerful mathematical tool for digital watermarking due to its ability to modify image characteristics without noticeable visual distortion. However, SVD-based methods alone lack resistance against analytical attacks and can be reverse-engineered to extract or alter hidden data. To overcome this limitation, incorporating entropy — a measure of randomness derived from live scenes — provides an effective enhancement. The entropy of a real-time environment introduces unpredictability, ensuring that each watermark is unique and non-reproducible.

This paper proposes an innovative system that combines SVD with live scene entropy for detecting fake documents. By embedding entropy-based watermarks dynamically, the proposed model enhances document security while maintaining computational efficiency. The approach effectively bridges the gap between invisibility, robustness, and real-time applicability for document verification.

II. Literature Review

Digital watermarking has been widely researched as a method to guarantee document authenticity and copyright protection. Initial research by Liu and Tan (2002) proposed a watermarking technique using Singular Value Decomposition (SVD), showing that hidden watermarks were possible to embed in images by making subtle changes to singular values. Although this technique provided simplicity and stealth, follow-up work by Wu (2005) found that explicitly tampering with singular values made the technique insecure. This was further confirmed by Naderahmadian (2015), who provided real-world attacks that effectively broke SVD-based watermarking schemes. These works together suggest that SVD watermarking is extremely vulnerable to adversarial tampering.

To counter these limitations, researchers have integrated statistical methods like Principal Component Analysis (PCA). A PCA-based authentication framework that enhanced watermark localization and attack resilience was suggested by Ding, Li, and Liu (2009). The greater security, however, came at the cost of computational complexity, thus reducing the suitability of such practices for resource-limited environments.

Entropy-driven approaches have also been attempted to increase invisibility of the watermark. Li and Chen (2008) employed entropy models for watermark embedding with negligible perceptual degradation but with transparency being

more important than robustness. Yassin, Salem, and Adawy (2012) integrated PCA with entropy in video watermarking for adaptive watermarking against compression and certain attacks. However, such a hybrid solution was demanding in terms of storage capacity and processing, rendering it unsuitable for light systems. Entropy-guided watermarking in the wavelet domain, balancing imperceptibility and robustness, was suggested by Hsieh and Tseng (2006). Even with these advances, computational overhead was an ongoing problem.

Overall, existing works reflect an evolutionary trend from straightforward SVD watermarking to hybrid schemes combining PCA, entropy, and wavelet transform. As each innovation promoted invisibility, robustness, or flexibility, none of them did so simultaneously within an efficient computation framework.

A. Research Gap

Simple SVD watermarking techniques are susceptible to analytical and practical attacks, which restricts their security. While PCA enhances watermark resistance, the resulting computational overhead limits real-time and large-scale applicability. Entropy-based watermarking guarantees invisibility but breaks down under robust manipulations like tampering, compression, or geometric deformations. Using PCA, entropy, and wavelet transforms in combination increases security but requires vast amounts of processing and storage capabilities. None of the methods described integrate external randomness to create time-specific, one-time digital signatures, making systems vulnerable to replication and forgery.

Overcoming these constraints, the work proposed here exploits live-scene entropy to create irreproducible digital signatures and inserts them through SVD-based watermarking. This unification is expected to deliver a tamper-proof, computationally feasible, and scalable system for protecting sensitive documents like certificates, identity certificates, and contracts.

III. Methodology

This section presents a novel entropy-based document watermarking framework that leverages Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) for robust watermark generation and embedding. The proposed system consists of four primary stages: entropy extraction, watermark generation, watermark embedding, and authenticity verification.

A. Entropy Extraction from Scene Photography

The watermark generation process begins with entropy extraction from a reference scene photograph, which serves as the unique signature source. Given an input photograph P of arbitrary dimensions, the image is first preprocessed to ensure consistency across different capture conditions.

For RGB images, conversion to grayscale is performed using MATLAB's `rgb2gray` function, which applies the standard luminance formula:

$$P_{\text{gray}} = 0.299R + 0.587G + 0.114B \tag{1}$$

The grayscale image is subsequently resized to a standard dimension of 256×256 pixels using bicubic interpolation to maintain spatial consistency across all processed images.

To capture local texture complexity, the image is partitioned into non-overlapping blocks of size 8×8 pixels, resulting in a 32×32 grid of blocks. For each block $B_{i,j}$ where $i, j \in \{1, 2, \ldots, 32\}$, the Shannon entropy is computed using MATLAB's `blockproc` and `entropy` functions:

$$H(B_{i,j}) = -\sum_{k=0}^{255} p_k \log_2(p_k) \tag{2}$$

where p_k represents the probability of occurrence of gray level k within the block. This block-wise entropy calculation produces an entropy map $E \in \mathbb{R}^{32\times32}$, which is then vectorized into a one-dimensional row vector $\mathbf{e} \in \mathbb{R}^{1\times1024}$ through column-wise flattening.

B. Watermark Generation via PCA Compression

The high-dimensional entropy vector $\mathbf{e}$ undergoes dimensionality reduction through Principal Component Analysis to generate a compact binary watermark. This compression step ensures computational efficiency while preserving the most significant variance in the entropy features.

The entropy vector is first zero-centered:

$$\mathbf{e}_c = \mathbf{e} - \mu_e \tag{3}$$

where $\mu_e = \text{mean}(\mathbf{e})$ denotes the scalar mean of the entropy vector.

To facilitate PCA, the 1024-element centered vector is reshaped into a data matrix. The vector is first trimmed to 1020 elements (the largest multiple of 10 that does not exceed 1024) and then reshaped into $\mathbf{E} \in \mathbb{R}^{102\times10}$, effectively treating the entropy values as 102 observations across 10 features:

$$\mathbf{E} = \text{reshape}(\mathbf{e}_c[1:1020], [102, 10]) \tag{4}$$

Standard PCA is then applied using MATLAB's `pca` function to decompose the data matrix:

$$[\mathbf{P}, \mathbf{T}, \boldsymbol{\lambda}] = \text{pca}(\mathbf{E}) \tag{5}$$

where $\mathbf{P}$ contains the principal component coefficients (loadings), $\mathbf{T}$ represents the score matrix (principal component projections), and $\boldsymbol{\lambda}$ contains the eigenvalues. We retain the first five principal components, corresponding to the directions of maximum variance, yielding a reduced representation $\mathbf{T}_5 \in \mathbb{R}^{102\times5}$.

The compressed feature matrix is vectorized through column-wise flattening, producing a vector of length 510:

$$\mathbf{f}_{\text{flat}} = \text{flatten}(\mathbf{T}_5) \in \mathbb{R}^{1\times510} \tag{6}$$

The first 50 elements are extracted to form the PCA feature vector:

$$\mathbf{f} = [\mathbf{f}_{\text{flat}}]_{1:50} \in \mathbb{R}^{1\times50} \tag{7}$$

Binary quantization is performed through zero-threshold comparison to generate the watermark bit sequence:

$$w_i = \begin{cases} 1, & \text{if } f_i > 0 \\ 0, & \text{otherwise} \end{cases}, \quad i = 1, 2, \ldots, 50 \tag{8}$$

This produces the final binary watermark vector $\mathbf{w} \in \{0,1\}^{1\times 50}$, which encapsulates the entropy characteristics of the original scene photograph in a compact form. A visual representation is created by reshaping the watermark into a 10×5 binary image.

C. SVD-Based Watermark Embedding

The generated binary watermark is embedded into the target document image using Singular Value Decomposition, a technique that modifies the intrinsic spectral properties of the image while maintaining visual quality.

Let $\mathbf{D} \in \mathbb{R}^{256\times 256}$ represent the preprocessed grayscale document image (converted from RGB if necessary and resized to standard dimensions). The SVD factorization yields:

$$\mathbf{D} = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{9}$$

where $\mathbf{U} \in \mathbb{R}^{256\times 256}$ and $\mathbf{V} \in \mathbb{R}^{256\times 256}$ are orthogonal matrices containing the left and right singular vectors respectively, and $\mathbf{S} \in \mathbb{R}^{256\times 256}$ is a diagonal matrix with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{256} \geq 0$ arranged in descending order.

Critically, the original singular value matrix $\mathbf{S}_{\text{original}}$ is preserved before modification, as it is required for watermark extraction during verification.

The watermark vector is first converted to a column vector $\mathbf{w}_{col} \in \mathbb{R}^{50\times 1}$. The watermark is embedded by modifying the first 50 diagonal elements of the singular value matrix through additive modulation:

$$\tilde{\sigma}_i = \sigma_i + \alpha \cdot w_i, \quad i = 1, 2, \ldots, 50 \tag{10}$$

where α is the embedding strength parameter that controls the trade-off between robustness and imperceptibility. In this implementation, $\alpha = 10.0$ was empirically determined to provide sufficient detectability while maintaining document readability.

The modified singular value matrix $\tilde{\mathbf{S}}$ retains all unmodified singular values for $i > 50$:

$$\tilde{S}_{i,i} = \begin{cases} \sigma_i + \alpha \cdot w_i, & i \leq 50 \\ \sigma_i, & i > 50 \end{cases} \tag{11}$$

The watermarked document is reconstructed using the modified singular values:

$$\mathbf{D}_w = \mathbf{U}\tilde{\mathbf{S}}\mathbf{V}^T \tag{12}$$

The reconstructed image is converted back to 8-bit unsigned integer format for storage. The system stores both $\mathbf{S}_{\text{original}}$ and the parameter α in a separate file (`original_S_values.mat`) to enable verification.

D. Document Authenticity Verification

The verification process extracts the embedded watermark from a potentially watermarked document and compares it against the original watermark to determine authenticity. The verification is performed without requiring access to the original unwatermarked document, relying instead on the stored original singular values.

Given a test document $\mathbf{D}_t \in \mathbb{R}^{256\times 256}$ (in double precision format), SVD decomposition is performed:

$$\mathbf{D}_t = \mathbf{U}_t\mathbf{S}_t\mathbf{V}_t^T \tag{13}$$

The watermark extraction compares the singular values of the test document against the stored original singular values $\mathbf{S}_{\text{original}}$ using the known embedding parameter α:

$$\tilde{w}_i = \frac{S_t(i,i) - S_{\text{original}}(i,i)}{\alpha}, \quad i = 1, 2, \ldots, 50 \tag{14}$$

where $S_t(i,i)$ and $S_{\text{original}}(i,i)$ denote the i-th diagonal elements of their respective singular value matrices.

To account for numerical precision and minor reconstruction errors inherent in the SVD process, binary quantization employs an adaptive threshold of 0.3 rather than 0.5:

$$\hat{w}_i = \begin{cases} 1, & \text{if } \tilde{w}_i > 0.3 \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

The extracted watermark $\hat{\mathbf{w}} \in \{0,1\}^{1\times 50}$ is reshaped from the column vector extraction result to a row vector for comparison against the reference watermark $\mathbf{w}$.

1) Similarity Metrics: Three complementary metrics quantify the correspondence between the original and extracted watermarks:

Bit Error Rate and Bit Accuracy: The number of incorrectly recovered bits provides a direct measure of extraction accuracy:

$$N_{\text{errors}} = \sum_{i=1}^{50} |\hat{w}_i - w_i| \tag{16}$$

The bit accuracy is computed as:

$$\text{Accuracy} = \frac{50 - N_{\text{errors}}}{50} \times 100\% \tag{17}$$

Normalized Cross-Correlation: The Pearson correlation coefficient measures linear similarity using MATLAB's `corr2` function:

$$\rho = \text{corr2}(\mathbf{w}, \hat{\mathbf{w}}) \tag{18}$$

Special handling is implemented for degenerate cases where either watermark has zero standard deviation, in which case the similarity is set to 1.0 if the watermarks are identical, and 0.0 otherwise.

Direct Similarity Score: The Hamming similarity provides an intuitive matching proportion:

$$\text{Similarity}_{\text{direct}} = \frac{1}{50}\sum_{i=1}^{50} \mathbb{1}(w_i = \hat{w}_i) \tag{19}$$

where $\mathbb{1}(\cdot)$ is the indicator function.

2) Authentication Decision: A document is classified as authentic if the bit accuracy exceeds a predefined threshold τ:

$$\text{Authentication} = \begin{cases} \text{AUTHENTIC}, & \text{if Accuracy} \geq \tau \\ \text{FAKE/TAMPERED}, & \text{otherwise} \end{cases} \tag{20}$$

In this study, $\tau = 90\%$ (corresponding to 45 out of 50 bits correctly matched) was established as the authentication threshold, providing a balance between security and tolerance for minor degradations from compression or processing.

E. Implementation Details

The complete watermarking system was implemented in MATLAB with a modular architecture consisting of five primary functions: `entropy_extraction.m`, `generate_watermark.m`, `embed_watermark.m`, `verify_document.m`, and the main execution script `main.m`. All image processing operations utilize MATLAB's Image Processing Toolbox, with SVD computations performed using the built-in `svd` function and PCA using the `pca` function from the Statistics and Machine Learning Toolbox.

The system generates several output files for analysis and verification: the watermarked document image, entropy map visualization, watermark visualizations (both original and extracted), the stored original singular values with embedding parameter, a detailed verification report, and a watermark comparison visualization. A dedicated `results` directory is created to organize all output files.

The workflow proceeds sequentially through four stages as orchestrated by the main script: (1) entropy extraction from the scene photograph producing a 1024-element feature vector, (2) PCA-based watermark generation creating a 50-bit binary signature, (3) SVD-based embedding into the document with storage of original parameters, and (4) verification through watermark extraction and multi-metric similarity assessment.

F. Results

The watermarking system was successfully applied using Singular Value Decomposition (SVD) to embed and recover watermark data from digital document images. The embedding process retained the visual quality of the document to be almost the same as in the original image, which is suggestive of the fact that the watermark was not visible to the human eye. The watermarks extracted from test images bore a great similarity to the original pattern of the watermark, thereby validating the accuracy and reliability of the system.

Quantitative assessment was conducted in terms of significant metrics—bit accuracy. The value of bit accuracy reflecting the percentage of accurately recovered watermark bits was always in excess of 90%, with all this implying very successful watermark recovery. The correlation coefficient of the original and extracted watermark also moved close to unity, thus ensuring that the embedded watermark pattern was preserved satisfactorily even after the reconstruction. The system was then tested using unwatermarked or attacked documents, where it was found that the output exhibited a marked difference in correlation and accuracy, proving that the method can satisfactorily distinguish between real and forged documents. Overall, the system proved efficient and accurate for watermark-based document authentication.

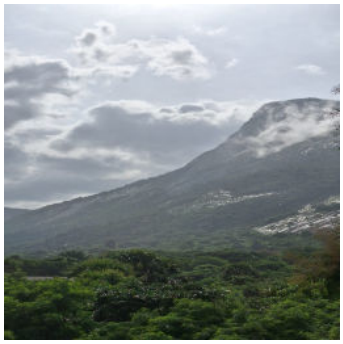

Fig. 1. Live scene image used to extract entropy[Size:256x256].

Fig. 2. Water mark embedded document[Size:256x256].

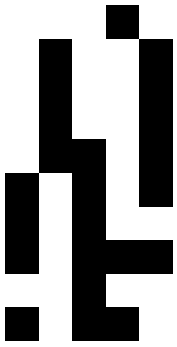

Fig. 3. The water mark generated[Size:10x5].

G. Limitations and Future scope

Despite the system showing promise, it does have some limitations. The current setup converts every input image to grayscale before applying the watermark. This reduces color details. As a result, the final image does not keep the document's original colors. Additionally, the embedding method works best with images of a fixed size (256×256). This limits its ability to handle documents of different sizes

or resolutions. The system also has trouble with geometric changes like rotation, cropping, or compression. These issues can affect how well the watermark is retrieved. Furthermore, the watermark capacity is limited, which restricts the amount of information that can be securely embedded. Currently, there is no encryption layer to protect the watermark data from extraction attempts.

In the future, the system could be improved to add watermarks to color images while keeping visual quality. The watermarking algorithm can be combined with other methods like Discrete Wavelet Transform (DWT) or Discrete Cosine Transform (DCT) to make it stronger and more resistant to noise or compression. Machine learning models could also be used to better detect tampering. Additionally, a flexible system could be created to handle images of different sizes and formats and include security measures to protect the watermark information. Developing a real-time authentication system for verifying official documents is another exciting opportunity for future development.

H. Conclusion

The success of Singular Value Decomposition (SVD) as a technique for safely and covertly embedding watermarks is shown in the project on document authentication using digital watermarking. The findings demonstrate that SVD-based watermarking provides a dependable method of confirming document authenticity without significantly altering the image's visual content. Using correlation analysis and bit accuracy, the extraction and verification procedure correctly identified if a document was edited or original.

Through this study, a stronger comprehension of the application of mathematical transformations to image processing and practical security challenges was gained. According to the study, watermarking can be a helpful technique for digital communication data protection and authentication.

REFERENCES

[1] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] K. Elissa, "Title of paper if known," unpublished.

[5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.