Chapter 2
Spatial Domain Watermarking

2,1 Introduction

Early research in video watermarking primarily focused on directly altering the pixel values in the video frames to embed the watermark, with minimal computational requirements. These methods are simple and less resource-intensive. Two main types of video watermarking can be performed in the spatial domain:

1. **Visible Watermarking**
2. **Invisible Watermarking**

2.2 Invisible Watermarking

Invisible watermarking refers to embedding a watermark in such a way that it remains imperceptible to the viewer, and only the knowledge of a decoding key allows the extraction of the embedded message. This method is typically used for proof of ownership. There are two main approaches to invisible watermarking in the spatial domain:

1. **Least Significant Bit (LSB) Substitution Approach**
2. **Correlation-Based Approach**

2.2.1 Correlation-Based Method

An alternative method for embedding a digital watermark in the spatial domain involves leveraging the correlation properties of noisy pseudo-random sequences, which are inherently additive in nature. These sequences are well-suited for watermarking due to their low amplitude, similar to noise, and their strong correlation characteristics that make them resistant to interference. The use of pseudo-noise (PN) sequences for watermarking is advantageous for the following reasons:

1. **Randomness**: PN sequences are inherently random and unpredictable.

2. **Seed-Based Generation**: These sequences are generated using a specific initial seed, which is known only to the sender and receiver.
3. **High Entropy**: The sequences undergo multiple stages of randomness, increasing their complexity.
4. **Security**: Without knowledge of the seed and the generation algorithm, it becomes highly challenging to predict or replicate the sequence.

In the proposed method, two distinct pseudo-random sequences are generated using the same key: one sequence is used when the watermark bit is set to 1, and the other is used when the watermark bit is set to 0. Figure 2.1 illustrates the process of generating the pseudo-random patterns used for watermark embedding.

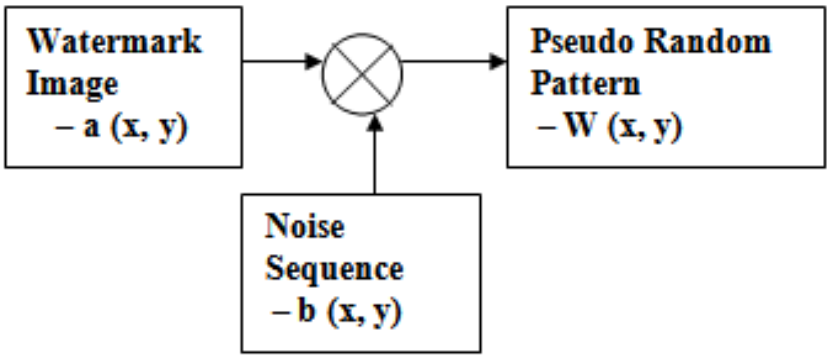


Figure 2.1: Generation of Pseudo Random Sequence

This method utilizes the inherent correlation of the PN sequences with the video content to ensure the watermark remains robust against various attacks while being imperceptible to the human eye.

2.3 Embedding Process:

Following is the sequential procedure used for the purpose of embedding a digital watermark into the video.

1. Original Video is broken into number of frames.
2. Faces are identified and located within the frame using algorithms such as Viola-Jones or deep learning-based detectors.
3. Block-size is decided based on the size of the Faces and Gain Factor is selected.
4. Two highly un-correlated PN sequence are generated using the key.
5. If message bit contains zero, PN sequence zero is added to that portion of watermark mask. Otherwise mask is filled with PN sequence one.

6. Add watermark mask to face image using gain factor K. Equation 2.1 shows the process.

$$\mathbf{fw}(\mathbf{x},\mathbf{y}) = \mathbf{f}(\mathbf{x},\mathbf{y}) + \mathbf{K} * \mathbf{W}(\mathbf{x},\mathbf{y}) \qquad (\mathbf{2.1})$$

Where

f (x, y) – Original Image Pixel

W(x, y) - Pseudo-random noise (PN) pattern

K - Gain Factor

fw (x, y) – Watermarked Image Pixel

7. Repeat the process until all the faces and frames are watermarked.
8. The watermarked face image, now containing the embedded watermark, is reinserted back into its original position within the image
9. Combine all frames to get the final watermarked video.

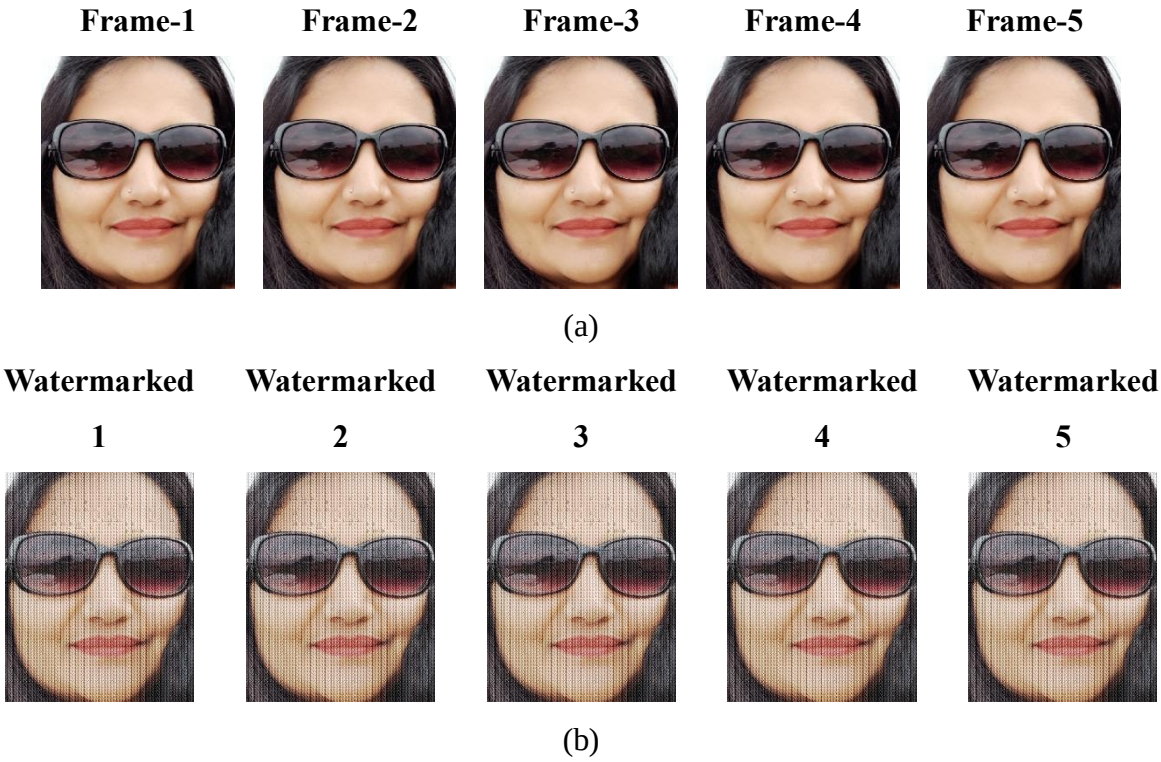


Figure 2.2: Example of Correlation based watermarking method with K = 100 (a) Five frames of video (b) Watermarked Frames

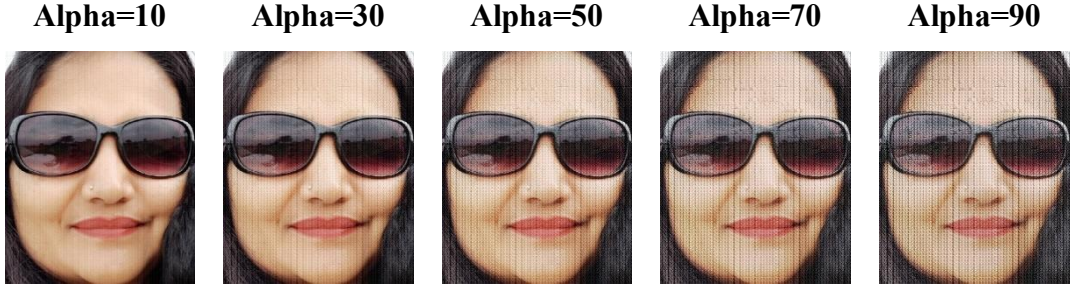
2.4 Extraction Process

The following outlines the step-by-step process employed to extract a digital watermark from a video:

1. **Decomposition of Video**: The watermarked video is split into multiple individual frames for processing.
2. **Face Detection**: Specialized face detection algorithms are utilized to locate and identify facial regions within each watermarked frame.
3. **Block Size Determination**: A suitable block size is selected for processing each frame.
4. **PN Sequence Generation**: Two highly uncorrelated Pseudo-Noise (PN) sequences are generated using a predefined key to serve as reference patterns.
5. **Frame Division**: Each frame is divided into multiple blocks based on the chosen block size.
6. **Correlation Calculation**: For every block in the frame, the correlation with each of the two PN sequences is calculated.
7. **Message Bit Retrieval**: If the correlation with the first PN sequence is greater than that with the second PN sequence, the corresponding message bit is assigned a value of 1. Otherwise, it is assigned 0. This process is repeated for all blocks in the frame to reconstruct the message.
8. **Processing Subsequent Frames**: The above steps are iteratively applied to each subsequent frame until the watermark from all frames is fully extracted.

Figure 2.3: Recovered Messages

Results



35.9899 **30.8038** **30.014** **29.6382** **29.4204**

(a)

Alpha=10 **Alpha=30** **Alpha=50** **Alpha=70** **Alpha=90**

0.6959 **0.916** **0.9751** **0.9828** **0.9828**

(b)

Figure 2.4: Various values of K (a) Frame 1-Watermarked (b) Recovered Messages

Frame No.	PSNR (db)	MSE	Correlation
1	29.1135	77.5368	0.9809
2	29.3435	75.6368	0.9840
3	29.4035	78.6068	0.9839
4	28.3435	75.3268	0.9800
5	29.5221	76.5068	0.9854

Table 2.1: Results with K=100 & BS=8

Table 2.2 Results with various frames with K=100.

Alpha	PSNR (db)	MSE	Correlation
10	35.9899	16.3715	0.6959
20	31.931	41.6847	0.8426
30	30.8038	54.0376	0.916
40	30.2978	60.7156	0.9603
50	30.014	64.8155	0.9751
60	29.7992	68.1027	0.9777
70	29.6382	70.6744	0.9828
80	29.5221	72.5882	0.9828
90	29.4204	74.3087	0.9828
100	29.3435	75.6368	0.9839

Table 2.2 Results with various values of K

Observation:

- **Perceptibility** decreases with a higher gain factor.
- **Robustness** increases with a higher gain factor.
- The watermark remains robust under certain attacks like cropping but vulnerable to others like rotation and filtering.

Chapter 3
Transform Domain Watermarking

Transform Domain Video Watermarking

The transform domain approach for video watermarking operates on the frequency representation of an image, contrasting with spatial domain techniques. Algorithms based on DCT and DWT are the most commonly used methods in this domain. These techniques transform an image from its spatial domain into the frequency domain, organizing frequency coefficients based on human perception. The coefficients are then modulated to embed watermark data, following three main steps:

1. **Forward Transformation**: Converts the spatial domain image into the frequency domain, producing frequency coefficients.
2. **Coefficient Modification**: Alters these coefficients based on the watermark information.
3. **Inverse Transformation**: Converts the modified frequency domain representation back into the spatial domain.

3.1 Discrete Cosine Transform (DCT)

The DCT converts a 2D spatial domain image into its frequency domain equivalent. It maintains the size of the transformed image equal to the original and positions the DC coefficient, representing low frequencies, in the top-left corner. The remaining coefficients, termed AC coefficients, increase in frequency along a zigzag path. The DC coefficient, always an integer, ranges from -1024 to 1023, while AC coefficients may be integers or non-integers. DCT's ability to distinguish between frequency components makes it an effective tool for watermarking. Most critical information is found in low-frequency components, while mid-frequency components are ideal for watermarking due to their balance of robustness and imperceptibility.

The ability of DCT to differentiate frequency components effectively makes it a highly useful tool for watermarking applications. The mathematical representations for two-dimensional DCT and its inverse are provided in Equations 1 and 2, respectively.

$$\mathbf{F(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\cos[\frac{(2x+1)u\pi}{2M}]\cos[\frac{(2y+1)v\pi}{2N}]} \quad --(3.1)$$