Dolley Shukla* and Manisha Sharma

A Novel Scene-Based Video Watermarking Scheme for Copyright Protection

DOI 10.1515/jisys-2017-0039

Received February 1, 2017; previously published online June 30, 2017.

Abstract: Many illegal copies of original digital videos are being made, as they can be replicated perfectly through the Internet. Thus, it is extremely necessary to protect the copyrights of the owner and prevent illegal copying. This paper presents a novel approach to digital video watermarking for copyright protection using two different algorithms, whereby successive estimation of a statistical measure was used to detect scene boundaries and watermark was embedded in the detected scenes with discrete wavelet transform. Haar wavelet was used for decomposition. For embedding, the approaches used were (i) the detailed subband (LH subband) and (ii) the approximate subband (LL subband) of the cover video. Imperceptibility, robustness, and channel capacity were measured using both algorithms. The system was tested for robustness in the presence of 15 different attacks of five different categories, and, under multiple attacks, ensured that a wide spectrum of attack analysis has been done. The performance metrics measured included mean square error, peak signal-to-noise ratio, structural similarity index, normalized correlation, and bit error rate. The experimental results demonstrated the better visual imperceptibility and improved performance in terms of normalized correlation and bit error rate with embedding using the LL subband. Comparative analysis with existing schemes proved the improved robustness, better imperceptibility, and reduced computational time of both the proposed schemes.

Keywords: Video watermarking, scene change detector, frame, histogram difference, robustness, watermark attack.

1 Introduction

Due to the fast expansion of the World Wide Web, the transmission, access, distribution, and saving of digital data have become very easy, and very time- and cost-effective over the Internet. However, with the help of several software tools, modification and perfect duplication/illegal copying of digital data have also become possible. The basic reason behind illegal copying is that there is no copyright or copy protection mechanism for transmitting data [27]. Hence, copyright and copy protection of content has become an important issue in the digital world. A permanent solution for this security issue of digital data is digital watermarking. Digital watermarking is the process of embedding watermark into the digital content that can be extracted later for ownership verification, i.e. copyright protection, and can be used for copy protection. For better security and effectiveness, imperceptibility, robustness, and embedding capacity are the main requirements of the digital watermarking system.

Depending on the type of data used for watermarking, there are four types of watermarking algorithms, i.e. image, text, video, and audio [16]. Researchers have dealt with image watermarking; however, video watermarking is the most focused area of researchers. Image watermarking algorithms are classified into spatial and frequency domains [21]. In the spatial domain watermark embedding schemes [6], watermark data are inserted by changing the pixel value of the gray level host image. The spatial domain watermarking techniques can be visible or invisible. Mostly, invisible watermarking is preferred for content authentication [20]. Again, the invisible kind of spatial watermarking can be of blind [14] or non-blind types. The qualities of the spatial watermark

*Corresponding author: **Dolley Shukla**, Associate Professor, Department of Information Technology, Shri Shankaracharya Technical Campus, Bhilai, Chhattisgarh, India, e-mail: dolley020375@gmail.com

Manisha Sharma: Department of Electronics and Telecommunication, Bhilai Institute of Technology, Durg, Chhattisgarh, India

are measured based on the embedding capacity of an image. It can be increased by embedding different bits of watermark image on different pixels of the cover image based on its color value [10]. The fidelity of an image can be improved by embedding based on the cover image prediction error sequence, and it can match very well with the properties of the human visual system [8]. Moreover, robustness can be achieved by embedding the watermark pixels on the most valuable area of an image called the region of interest (ROI) [25]. A big drawback that exists in the spatial domain is that spatial domain watermarks can be manipulated by some geometrical attacks. Therefore, frequency domain watermarking was introduced, which means transferring of the digital multimedia content into multiple frequency bands using reversible transforms, and then embedding is performed on the transformed coefficients, which are more robust against various image processing, video processing, and geometrical attacks. Some hybrid techniques have been discussed, such as discrete fourier transform (DFT) and Radon [23]; discrete cosine transform (DCT) and singular value decomposition (SVD) [36]; discrete wavelet transformation (DWT) and SVD, DWT and Hilbert transform [3]; and DCT, SVD, and ridgelet transform (RT) [5]. DWT, DCT, and SVD [37, 38] are used for medical image watermarking for embedding two image watermarks (Symptoms and Record) simultaneously in the LH2 and LL3 subbands of cover medical images. DWT and DCT [29] are suitable for protection of patient identity and for secure medical document dissemination over an open channel. DWT, DCT, and SVD, using encryption for teleophthalmology, fusion of discrete wavelet transforms and SVD, are used for simultaneous embedding of four different watermarks in the form of image and text in the non-ROI part of the eye as cover image [24]. The hybrid technique of Singh [32] improved the robustness, visual quality, capacity, and security of the watermarks; however, it increased the computational complexity.

Video watermarking differs from image watermarking. First, video signals are highly susceptible to pirate attacks, such as interpolation, frame swapping, frame dropping, frame averaging, etc. These attacks have no counterpart in image watermarking. Second, providing imperceptibility of the watermark on a video is relatively more difficult than on an image, because the watermark embedding procedure should consider the temporal variation into account due to the three-dimensional characteristics of the video. The third issue in video watermarking depends either on embedding the identical watermark in each frame, where an attacker would collude the frames from different scenes for extracting the watermark [35], which leads to the statistical perceptual invisibility maintenance problem or embedding independent watermark for each frame, where the attacker would take advantage of the motionless regions in successive video frames to remove the watermark by comparing and averaging the frames. The solution to the above-mentioned collusion and averaging problems pointed out by Su et al. [33] is embedding identical watermarks to the motionless frames and different watermarks to the motion frames. Thus, two types of watermarks can be embedded in the same video.

To increase the robustness, the watermark is embedded into the detail coefficient of wavelet transform [17]. The decomposed watermark is embedded into the decomposed video according to the decomposition level by Niu and Sun [22]. Serdean et al. [28] proposed a blind video watermarking scheme in the wavelet domain using the human visual system model. Barni et al. [4] proposed a DFT-based watermarking scheme. The scheme was robust against sharpening, scaling, JPEG compression, rotation, cropping, and filtering attacks.

Contourlet transform-based robust watermarking methods were also proposed [13]. Ranjbar et al. [26] proposed a blind and a robust watermarking method consisting of two embedding stages. In the first stage, the odd description of the image is divided into non-overlapping fixed-sized blocks, and a signature (watermark) is embedded in the high-frequency components of the contourlet transform (CT) of the blocks. In the second stage, the signature is embedded in the low-frequency components. However, this method is less resistant to median filtering, Gaussian noise, salt and pepper noise, and JPEG compression attacks. Agilandeewari and Ganesan [1] proposed a robust color video watermarking scheme based on hybrid embedding techniques. Embedding only the slices (not an entire image) will improve the level of imperceptibility, and is suitable for only copyright protection.

1.1 Contribution of the Work

On the basis of a literature survey, we are concerned about designing a robust digital video watermarking system based on DWT. To increase the imperceptibility and robustness, the watermark is embedded only

in selected frames. The selected frames are the frames where the scene change occurs. Therefore, detection of the accurate scene transition is the main goal. The scene change detector detects the accurate scene-changed frames using the successive histogram difference method. Using the same scene detection method, two schemes are proposed. Both the proposed schemes achieve good level of watermark quality with good peak signal-to-noise ratio (PSNR) values. Here, as the embedding of the watermark is done only in scene-changed frames using low- and high-frequency DWT subbands, it is robust against image processing attacks, geometrical attacks, JPEG compression, video attacks with high normalized correlation values, and low bit error rate (BER). Comparative analysis between two algorithms is also performed.

2 Proposed Methodology

Nowadays, a wavelet transform is a handy tool in the toolbox of every engineer doing research work in digital image processing and signals analysis. DWT gives a digital signal more information about the time, space, or frequency domain. In our watermarking scheme, the watermark is embedded in the DWT domain of the video frame. To improve the performance, the proposed approach was developed by combining the watermarking scheme and the scene change detector. To reduce the computational complexity and time, a watermark is embedded only into frames where abrupt scene change occurs. The system framework of the proposed approach is illustrated in Figure 1. In our watermarking scheme, a watermark is embedded in the DWT domain of a video frame. Successive estimation of statistical measure (SESAME) is used to detect abrupt scene changes. Histogram-based SESAME and HiBisLI method detect accurate abrupt scene changes. A watermark image is embedded into the scene-changed frame using either LL or LH subbands of the cover video. Haar wavelets are used for decomposition. On the basis of the decomposition level used, two different novel algorithms for embedding and extraction are proposed. The imperceptibility and robustness of both algorithms are tested under different attacks.

2.1 Scene Change Detector

The scene change detector detects abrupt scene changes using SESAME, i.e. histogram difference between the frames. To further enhance the result, the method is applied up to four levels. The histogram method is a

Figure 1: Scene-Based Video Watermarking System Using DWT.

process in which a signal is sorted out into various bins arranged in increasing order of magnitude of signal strength. The whole idea is to determine the spread of the signal over the whole spectrum and to use this spread data for further analysis. In this case, the difference of distance between histogram heights of the same bins is used as a statistical measure. To remove similar frames, a filtering method is used in the initial stage that filters out the same frames. To filter out the same frame, the filtering method consists of the histogram, binary search, and linear interpolation. This method is called the HiBisLI method.

2.2 Watermark Construction

A properly designed watermark is required for an efficient watermarking system that most easily adapts the cover data and must give better robustness under the degradation of perceptual quality. In the proposed watermarking system, the color image is taken as the watermark image. Then, the image is converted into gray scale image of size $N \times N$, i.e. 256×256 .

2.3 Watermark Embedding Algorithm

For the embedding process, video data are divided into frames. The discrete wavelet transforms using Haar wavelets are applied on that frame where scene change occurred. Discrete wavelet transform separates the image into four components: lower resolution approximation (LL), vertical (LH), horizontal (HL), and diagonal (HH). The designed system embeds the watermark image into one of the LH and LL subbands. Two algorithms are proposed, first for the embedding using the LH subband and second for the embedding using the LL subband. Both of the algorithms are explained below. The process is explained in detail as follows.

2.3.1 Embedding Using the LH Subband (Algorithm 1)

Watermark Embedding Algorithm Using the LH Subband (Algorithm 1)

Input: Cover video with frame size $M \times M$ and watermark image of size $N \times N$.

Output: watermarked video.

Step 1: Choose an appropriate watermark image of size $N \times N$ and a video as a cover video.

Step 2: Apply a preprocessing step on watermark image and convert the watermark image to gray image using an RGB to gray converter.

Step 3: Resize the watermark image into 128×128 .

Step 4: Apply preprocessing on cover video and convert into frame using a video to frame converter.

Step 5: Apply the scene change detector algorithm using SESAME based on the histogram difference method.

Step 6: Convert the scene-changed frame into gray level by applying an RGB to gray converter.

Step 7: Resize the scene-changed cover video frames into 256×256 size.

Step 8: Decompose the frame where scene change occurs, by applying one-level 2D-DWT on that scene-changed frame. The frame is converted into four subbands (LL, LH, HL, and HH) of size $M/2 \times M/2$, i.e. (128×128 size), three details, and one approximation.

LL – The approximation looks just like the original. All the energy is contained in the LL subband.

LH, HL, and HH – Detail subband where LH & HL band preserves localized horizontal & localized vertical features and HH band isolate localized high-frequency point features in the original video frame.

HH – The high-frequency components are usually used for watermarking as the human eye is less sensitive to changes in edges.

Step 9: The watermark image is inserted into the cover video using the α blending technique. In this technique, the decomposed components of the cover video frame where scene change is detected and the watermark are multiplied by a scaling factor and are added. We have taken $\alpha = 0.05$.

For embedding in the LH part – $ELL = VLL$;

$ELH = VLH + \alpha * \text{img_resized_to_vlh}$; $HL = VHL$;

$EHH = VHH$;

where VLL, VLH, VHL, and VHH – low-frequency and high-frequency approximation of the cover video;

where α represents the embedding factor.

Step 10: After embedding the watermark image on the scene-changed cover video frames, the inverse DWT is applied to the watermarked video frames to generate the final secure watermarked image.

2.3.2 Embedding Using the LL Subband (Algorithm 2)

Watermark Embedding Algorithm Using the LL Subband (Algorithm 2)

Input: Cover video with frame size $M \times M$ and watermark image of size $N \times N$.

Output: watermarked video.

Step 1: Choose an appropriate watermark image of size $M \times N$ and a video as a cover video.

Step 2: Apply a preprocessing step on the watermark image and convert the watermark image to gray image using an RGB to gray converter.

Step 3: Resize the watermark image into 128×128 .

Step 4: Apply preprocessing on the cover video and convert into frames using a video to frame converter.

Step 5: Apply the scene change detector algorithm using SESAME based on the histogram difference method.

Step 6: Convert the scene-changed frame into gray level by applying an RGB to gray converter.

Step 7: Resize the scene-changed cover video frames into 256×256 .

Step 8: Decompose frame where scene change occurs by applying one-level 2D-DWT on that scene-changed frame. The frame is converted into four subbands (LL, LH, HL, and HH) of size $M/2 \times M/2$, i.e. (128×128 size), three details, and one approximation.

LL – The approximation looks just like the original. All the energy is contained in the LL subband.

LH, HL, and HH – Detail subband where LH & HL band preserves localized horizontal & localized vertical features and HH band isolate localized high-frequency point features in the original video frame.

HH – The high-frequency components are usually used for watermarking as the human eye is less sensitive to changes in edges.

Step 9: The watermark image is inserted into the cover video using the α blending technique. In this technique, the decomposed components of the cover video frame where scene change is detected and the watermark are multiplied by a scaling factor and are added. We have taken $\alpha = 0.05$.

For embedding in LL part – $ELL = VLL + \alpha * \text{img_resized_to_vll}$;

ELH = VLH;

EHL = VHL;

EHH = VHH;

where VLL, VLH, VHL, and VHH – low-frequency and high-frequency approximation of the cover video;

where α represents the embedding factor.

Step 10: After embedding the watermark image on the scene-changed cover video frames, the inverse DWT is applied to the watermarked video frames to generate the final secure watermarked image.

2.4 Watermark Detector and Extraction Algorithm

The extraction is just the reverse operation of embedding and comprises three steps, i.e. watermarked video preprocessing and detection, extraction, and watermarked video postprocessing. Firstly, the watermarked video is converted into frames. The presence of watermark is detected with the verification of the scene change in the frame. If there is a scene-changed frame, then it shows that “watermark is present”. An extraction is the inverse operation of embedding. By performing the subtraction operation between the particular subband of the watermarked video frame and a cover video frame, the watermark image is extracted. The particular subband in which the embedding is performed is chosen. Ownership or copyright protection is proven by extraction of the watermark from a particular scene-changed watermarked frame.

This process is explained in detail as follows.

2.4.1 Extraction using the LH Subband (Algorithm 1)

Watermark Extraction Algorithm Using the LH Subband (Algorithm 1)

Step 1: Consider the watermarked video.

Step 2: Apply preprocessing on the cover video and convert into frame using a video to frame converter.

Step 3: Apply the scene change detector algorithm using SESAME based on the histogram difference.

Step 4: Convert the scene-changed frame into gray level by applying an RGB to gray converter.

Step 5: Resize the scene-changed watermarked video frames into 256×256 .

Step 6: Apply DWT on the scene-changed watermarked video frame, which decomposed the image into four subbands.

Step 7: Apply α blending on LH frequency components that are used for the embedding process.

$ILL = (ELH - CLH) / \alpha$;

where

ILL – extracted/recovered watermark image from the low-frequency approximation of the embedded video;

ELL – low-frequency approximation of embedded watermarked video frame;

CLL – low-frequency approximation of the cover video frame.

Step 8: The extracted image is resized and converted into normal form using uint 8.

2.4.2 Extraction Using the LL Subband (Algorithm 2)

Watermark Extraction Algorithm Using the LL Subband (Algorithm 2)

Input: Watermarked video.

Output: Extracted image.

Step 1: Consider the watermarked video.

Step 2: Apply preprocessing on the cover video and convert into frame using a video to frame converter.

Step 3: Apply the scene change detector algorithm using SESAME based on the histogram difference.

Step 4: Convert the scene-changed frame into gray level by applying the RGB to gray converter.

Step 5: Resize the scene-changed watermarked video frames into 256×256 .

Step 6: Apply DWT on the scene-changed watermarked video frame, which decomposed the image into four subbands.

Step 7: Apply α blending on LL frequency components that are used for the embedding process.

$$ILL = (ELL - CLL)/\alpha;$$

where

ILL – extracted/recovered watermark image from the low-frequency approximation of the embedded video;

ELL – low-frequency approximation of the embedded watermarked video frame;

CLL – low-frequency approximation of the cover video frame.

Step 8: The extracted image is resized and converted into normal form using uint 8.

3 Results and Discussion

To evaluate the performance of the proposed schemes, experiments are performed in the MATLAB 2015 platform on five .avi videos having different frame sizes and frame rates (i.e. Ad, NEWS, Sports, Children, and Documentary), and five different images (i.e. Cameraman, Lena, Mandril, Pepper, and Finger) of size 256×256 are considered. The designed system performs the embedding process using two different methods, i.e. subbands of DWT: (i) LL subband and (ii) LH subband.

In both methods, the embedding process is performed by using DWT coefficients of the scene-changed frame of the cover video and watermarked image. The watermarked image is invisible to the human eye. There is no remarkable difference between the cover video and watermarked video. For invisible watermarking, the embedding parameter is selected as 0.05. The effectiveness of the proposed scheme has been shown by verifying the proposed scheme against various experiments in terms of (i) imperceptibility, (ii) robustness, and (iii) embedding capacity.

3.1 Scene Change Detector

The outputs of the system for the scene change detector using histogram-based SESAME and HiBisLI are compared against FFMPEG as a reference, displayed in Figure 2. Precision, recall, and computational time are the performance evaluation parameters and calculated with reference to FFMPEG software output, shown in Table 1.

3.2 Imperceptibility

The alteration in the perceptual quality of the watermarked video should be determined. To measure imperceptibility, PSNR, mean square error (MSE), and structural similarity index (SSIM) are the main parameters. For optimized imperceptibility, the minimum acceptable value of PSNR is 38 dB, as suggested by Petitcolas.

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right), \quad (1)$$

Figure 2: (A) Scene Change Detected Frame Using Histogram-Based SESAME and HiBisLI (B) FFMPEG Output.

Table 1: Performance Evaluation of Scene Change Detector.

Video	No. of Scenes Detected	No. of Scenes (FFMPEG)	No. of Actual Scenes	No. of Missed Scenes Detected	No. of Missed Scenes Detected	Precision	Recall	Computational Time
Ad.avi	32	22	18	04	14	81.8%	56.25%	20.6235 s

$$\text{MSE} = \frac{1}{(m * n)} \sum_{i=1}^m \sum_{j=1}^n (W(i, j) - w'(i, j)), \quad (2)$$

where $W(i, j)$ and $w'(i, j)$ are the gray levels of pixels in the cover video frame and watermarked video frame.

$$\text{SSIM}(X, Y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (3)$$

where μ_x and μ_y are the average of X and Y , respectively; σ_{xy} is the covariance of X and Y ; μ_x^2 and μ_y^2 are variance of X and Y ; and C_1, C_2 are two variables.

3.2.1 Encoder Output

The performance of the encoder using Algorithms 1 and 2 is evaluated using the MSE and PSNR parameters, shown in Tables 2 and 3. Ad.avi with Pepper.jpg shows high PSNR value with Algorithm 1, and Ad.avi with Mandril.jpg

gives high PSNR value. To measure imperceptibility, SSIM is calculated between the original cover video frame and the watermarked video frame. For the comparison, the first frame is considered and represented in Figure 3.

3.2.2 Imperceptibility in Terms of SSIM

The perceptual quality of the watermarked video with and without an attack is measured in terms of SSIM. It is clear from Table 2 that the scene-based watermarking using a SESAME (histogram difference)-based method gives high PSNR and low MSE values. A comparative performance analysis of the system with both algorithms shows that the watermarking using LL subband gives better results. The similarity index is measured between the original video and the watermarked video. There is no remarkable difference between the two in both cases. It shows that the system is imperceptible, depicted in Figure 3. The empirical result of the comparative analysis of watermarking with LH subband (Algorithm 1) and with LL subband (Algorithm 2) shows 28.49% improvement in MSE and 21.52% improvement in PSNR value with Algorithm 2. The comparative analysis results between the two algorithms in terms of SSIM are represented graphically in Figure 4 and Tables 2 and 3.

Table 2: Performance Evaluation of Encoder for Imperceptibility Using the LH Subband (Algorithm 1).

Cover Video – AD.avi, Wavelet – Haar						
Cover video.avi and Watermark Image	MSE		PSNR		SSIM	
	LH Subband	LL Subband	LH Subband	LL Subband	LH Subband	LL Subband
Ad.avi and cameraman.jpg	0.310059	0.0221719	53.21634	64.672765	1	1
Ad.avi and mandril.jpg	0.1490700	0.0063117	56.396800	70.129312	1	1
Ad.avi and lena.jpg	0.1411837	0.1411840	56.632900	56.632950	1	1
Ad.avi and pepper.jpg	0.1124082	0.0166890	57.622821	65.906289	1	1

Performance of the system with LH Subband shows max. PSNR(57.622821 dB) and min. MSE using Ad.avi video with Pepper.jpg, image and with LL subband, Ad.avi video with mandril.jpg shows max. PSNR(57.622821 dB) and min. MSE value, represented in bold form in Table 2.

Table 3: Comparative Analysis of Algorithms 1 and 2.

Subband	NC	BER
LH subband	0.8136438	0
LL subband	0.924707	0

Figure 3: Imperceptibility First Frame of Watermarked Video: (A) Algorithm 1; (B) Algorithm 2.

Figure 4: Comparative Performance Analysis of Encoder Algorithms and Algorithm 2 (Imperceptibility Measurement). (A) MSE; (B) PSNR; and (C) SSIM.

3.3 Robustness

Robustness means the watermarked image should resist against different attacks. The robustness property of the proposed schemes are measured by using the parameters normalized cross-correlation (NC) and BER, and are measured by comparing the original watermarked image and the extracted watermarked image (without attack/after the attack). The range lie between -1 and $+1$. The NC value is approximately 1 if the watermarked image is almost similar to the original image, while negative watermarking gives -1 NC value. It becomes totally unacceptable or uncorrelated if the NC value tends to 0 BER, and NC can be calculated as follows [9].

NC: NC is used to compare the original watermark and extracted/recovered watermark from the watermarked video. NC can be derived using mathematical representation given below:

$$NC = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [W(i,j) * W'(i,j)]}{\text{sqrt} \left[\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (W(i,j)^2) \right] \text{sqrt} \left[\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (W'(i,j)^2) \right]}, \quad (4)$$

where M and N represent the width and height of the watermark image; $W(i, j)$ = pixel intensity value at coordinates i, j of the original watermark image; and $W'(i, j)$ = pixel intensity value at coordinates i, j of the extracted/recovered watermark image.

BER: It is the ratio of wrongly extracted watermark bits to the total number of watermark bits embedded. If there is no error in the received message, then the BER value will be 0; otherwise, it is close to 1. It can be computed using the following equation:

$$BER = \frac{\sum_{i=0}^{M-1} |W_i - W'_i|}{m} = \frac{\text{No. of error bits}}{\text{Total no. of embedded watermark bits}}, \quad (5)$$

where W_i = intensity of the i th pixel in the original watermark image; W'_i = intensity of i th pixel in the extracted watermark image; and m = total no. of embedded watermark bits.

A Original watermark images – Lena, Cameraman, Pepper.jpg **B** Extracted image – Lena, Cameraman, Pepper.jpg [LH Subband] **C** Extracted image – Lena, Cameraman, Pepper.jpg [LL Subband] **Figure 5:** (A–C) Original and Extracted Images of Lena.jpg, Cameraman.jpg, and Pepper.jpg.

3.3.1 Attack Analysis

The robustness of the proposed algorithms and its fidelity are tested on the sampled video (AD.avi, Sports.avi, etc.) using the following attacks: (i) image processing attack, (ii) geometrical attack, (iii) JPEG compression, and (iv) video attack.

3.3.1.1 No Attacks

If a watermark is embedded into a cover video, then its effectiveness may be considered in terms of transparency and robustness, and can be evaluated effectively using different parameters. Transparency can be measured using PSNR and MSE, and NC and BER are the metrics for robustness measurement. With no attack, the PSNR value is given as 53.21634 dB (Algorithm 1) and 64.6727 dB (Algorithm 2) for AD.avi video, and Cameraman.jpg is considered as the watermark image. Figure 5 shows the original and extracted outputs with NC values of both the algorithms with the no-attack condition.

The experimental result of the comparative analysis of watermarking with LH subband (Algorithm 1) and with LL subband (Algorithm 2) shows 13.65% improvement in normalized correlation value with Algorithm 2 as depicted in Figure 6 and represented in Table 3.

3.3.1.2 Different Attacks

To validate the performance of the proposed scheme, the system is tested under various attacks, i.e. image processing attacks, geometrical attacks, JPEG compression, and various video attacks. The performance of the proposed algorithms is evaluated in terms of imperceptibility and robustness against various signals. The entire system is developed in MATLAB R 2015a 64-bit version platform and run on a Dell Inspiron System Intel® Core™ i5 CPU @ 2.53 GHz processor and 4GB RAM with Windows 7, 64-bit operating system and X 64-based processor, USA. The proposed method's performance is measured in terms of imperceptibility, robustness, and channel capacity. Five sample videos of different frame sizes and frame rates as cover video, and five different images of the same size are taken to validate the performance. The sample videos are Documentary.avi, NEWS.avi, Ad.avi, sports.avi, and children.avi. The result is shown over the watermark image, i.e. Pepper.jpg.

The quality of the watermarked video is measured by MSE and PSNR. However, the robustness of the extracted watermark image is measured by using NC and BER under 15 different attacks. The visual effect of Gaussian noise on watermark with variance 0.05, and salt and pepper noise with noise density 0.05, seriously affects the extracted watermark quality (NCC, BER), which is reflected in our experimental results of Table 4. Under image processing attack, watermarking with LL subband sustains Gaussian low-pass filter (LPF) more effectively with NC value of 0.876779 and LH subband with sharpening 0.807959. Under a geometrical attack, the LL subband performs well with a stretching attack (NC value 0.9028592), with LH subband resizing (NC value 0.84913205), and JPEG compression values with LL subband 0.9301982 and LH subband 0.83514266. Under a video attack, both algorithms perform better with a frame dropping attack. The BER values for resizing and stretching are 0.000 and 0.00. In both the proposed algorithms, 20 frames are dropped randomly out of 1501 frames, and then we try to extract the watermark from the frame dropped watermarked video. Both the algorithms show minimum BER values for Gaussian LPF with LL subband 0.0557059 and with LL subband 0.096892. The experimental result proves that the LL subband is a suitable location for embedding.

The proposed system's ability of resisting different geometrical attacks and compression attacks in terms of NC and BER is shown in Tables 5–9. Table 5 demonstrates that the robustness of the system decreases with increasing rotation angle. The robustness improves with quality factor, as depicted in Table 7. Frame dropping is the process of dropping the frames from a video randomly. In both the proposed algorithms, different frames are dropped randomly out of 1501 frames and then we try to extract

Table 4: Comparative Analysis between Watermarking Using the LL and LH Subbands.

Attack Category	S. No.	Attack	NC		BER	
			LH Subband (Algorithm 1)	LL Subband (Algorithm 2)	LH Subband (Algorithm 1)	LL Subband (Algorithm 2)
Without attack	1	Without attack	0.8136439	0.9247070	0.0000000	0.0000000
Image processing attack	2	Salt and pepper noise (noise density 0.05)	0.7307824	0.7395973	0.3228410	0.3278247
	3	Gaussian noise	0.3414270	0.3514260	0.3751062	0.3776421
	4	Speckle noise (0.05 noise variances)	0.7649090	0.8368500	0.2597871	0.2650219
	5	Gaussian LPF	0.7952060	0.8767790	0.0968929	0.0557059
	6	Blurring (size 5 and sigma)	0.6351426	0.7600767	0.2956995	0.1619779
	7	Sharpening	0.807959	0.7980001	0.2486626	0.1226918
	8	Normal blur (radius 10)	0.6351426	0.7522937	0.3335858	0.2728159
	9	Motion blur	0.6514266	0.6919266	0.3262806	0.2544478
	10	Rotation (1°)	0.6351426	0.7265135	0.2664159	0.1938107
Geometrical attack	11	Resizing (by 1.05)	0.8491320	0.8417642	0.0000000	0.0000000
	12	Stretching (1.05*width)	0.8160824	0.9028592	0.0000000	0.0000000
JPEG compression	13	JPEG compression (image quality 40)	0.8351426	0.9301982	0.4356705	0.4379051
Video attack	14	Frame averaging	0.4902788	0.5733700	0.3661736	0.3619456
	15	Frame dropping (25 frames randomly)	0.8122048	0.9149219	0.0000000	0.0000000
	16	Frame swapping	0.4891468	0.3514265	0.3132254	0.2912059

Figure 6: Comparative Analysis of Decoder Performance of Algorithms 1 and 2.

Table 5: Robustness against Rotation Attacks.

Attack	LL Subband		LH Subband	
	NC (LL)	BER	NC (LH)	BER
Rotate 1°	0.890127	0.193810	0.871606	0.266415
Rotate 2°	0.872065	0.195810	0.860735	0.276425
Rotate 5°	0.863362	0.199383	0.850151	0.284159
Rotate 10°	0.847023	0.203810	0.820013	0.296425

Table 6: Robustness against Resizing Attacks.

Attack	LL Subband		LH Subband	
	NC (LL)	BER	NC (LH)	BER
Resize 1.05	0.841764	0.000	0.849132	0.0000
Resize 1.2	0.852176	0.0000	0.853132	0.0000
Resize 0.5	0.854176	0.0000	0.859132	0.0000
Resize 1.5	0.841764	0.0000	0.849132	0.0000

Table 7: Robustness against JPEG Compression Attacks.

Attack	LL Subband		LH Subband	
	NC (LL)	BER	NC (LH)	BER
JPEG compression ($Q=40$)	0.93019	0.124379	0.83514	0.14356
JPEG compression ($Q=50$)	0.93598	0.134379	0.84621	0.13356
JPEG compression ($Q=50$)	0.94298	0.144379	0.86534	0.11356
JPEG compression ($Q=60$)	0.93229	0.114379	0.87514	0.10356

the watermark from the frame-dropped watermarked video. The robustness against frame dropping is depicted in Table 8.

Despite all the abovementioned attacks, a new type of attack, i.e. the occurrence of more than one attack at the same time on the video in a one-by-one manner on all the frames, called multiple attacks, is introduced. Two combinations, (i) Frame Drop 25 + rotate 1° and (ii) Frame Drop 25 + rotate 10°, are used. Table 9 shows the robustness under two multiple attacks in terms of NC and BER, and proves that watermarking with the LL subband performs better.

Table 8: Robustness against Frame Dropping Attack.

Attack	LL Subband		LH Subband	
	NC (LL)	BER	NC (LH)	BER
Frame drop (4 frames)	0.944921	0.10237	0.85220	0.16437
Frame drop (25 frames)	0.914921	0.15379	0.81220	0.18437
Frame drop (41 frames)	0.954921	0.10043	0.83220	0.11413

Table 9: Robustness against Multiple Attacks.

Attack	LL Subband		LH Subband	
	NC (LL)	BER	NC (LH)	BER
Frame drop 25 + rotate 1°	0.890127	0.193810	0.87160	0.26641
Frame drop 25 + rotate 10°	0.847023	0.20381	0.820013	0.29642

3.4 Embedding Capacity or Payload

The maximum amount of data embedded to extract the watermark effectively in the receiver side, without affecting imperceptibility of original data, is known as the capacity of watermark or watermark payload. Increasing the embedded watermarking data capacity enhances the watermarking scheme robustness, which may affect the imperceptibility of the watermarked data.

In both the proposed schemes, a cover video with frame size 256×256 is considered. For one-level DWT, the cover video frame decomposed into four subbands of size 128×128 . The capacity of the algorithm is able to embed a payload size of 128×128 pixels, which means it is able to hide a total of 16,384 bits using 32 frames. The watermark image is inserted into the one subband of the cover video frame. Therefore, for a given N , one can embed a maximum of

$$\frac{N' \cdot \text{Watermark image size}}{N \cdot \text{Cover frame image size}}, \quad (6)$$

where N = no. of frames in a video: $[N \cdot (128 \times 128)] / [N' \cdot (256 \times 256)] = 0.25 \cdot [N/N']$ bits per pixel.

3.5 Computational Complexity

The complexity of video-based watermarking is one of the potential issues. Both the proposed schemes are scene-based watermarking, where watermark is embedded only into those frames where scene change occurs. Both the proposed techniques consist of two parts, i.e. scene change detection process and watermark embedding and extraction process. The computational complexity of the system is defined by time complexity and space complexity.

Time complexity: The total time complexity is the addition of the time complexity of both the processes given by [total time complexity: $(T_1 + T_2)$]. The scene change detection process uses SESAME with time complexity $O(n)$ and HiBiSLI algorithm that includes three algorithms – histogram, binary search, and linear interpolation – having $[O(n) + O[\log_2(n)] + O(n)]$. For watermark embedding and extraction, DWT is performed with time complexity $O(n)$. Therefore, the total time complexity of the proposed algorithms is $[O(n) + O[\log_2(n)]]$, i.e. approximately $[O(n)]$. This shows that the computational time complexity of both the proposed algorithms is linear in terms of the number of frames.

In the proposed algorithms, the AD.avi tested video consists of 1501 frames and only 32 frames are detected as scene-changed frames using the histogram-based SESAME method. The watermark embedding only in the scene-changed frame reduces the computational complexity and computational time. Complexity may further be reduced by using only one subband for embedding, i.e. either the LL subband or the LH subband. Comparative analysis of computational time is presented in Table 10. It is clear from Table 10 that the computational time of Algorithm 1 reduces by 79.37% and that for Algorithm 2 (watermarking with LL subband) reduces by 80.38%.

Space complexity: The space complexity of an algorithm is the maximum amount of space used at any one time, ignoring the space used by the input to the algorithm. In the proposed algorithms, 32 frames of frame size 360×288 are extracted. Therefore, the memory requirement for scene-based watermarking is $[32 \times (360 \times 288)]$, i.e. 3.31 MB for both the algorithms, whereas it is $[1501 \times (360 \times 288)]$, i.e. 155.62 MB, if all the frames used for embedding. Hence, both the proposed algorithms reduce the space complexity.

The embedding time per frame and the extraction time per frame for Algorithm 1 are 0.5514 and 0.12735 s, respectively. For Algorithm 2 (watermarking with the LH subband), the embedding time per frame and the extraction time per frame are 0.5195 and 0.123 s, respectively, represented in Table 11. Both the proposed schemes run on a general-purpose processor, i.e. Intel i5 processor. If algorithms run on specific processors, DSP pipeline architecture, or through programming optimization techniques, the speed can definitely be improved. In the present state, both the proposed algorithms are not for real-time applications; however, with the reduction of processing time, it may be used for real-time application.

4 Relationship between Watermark Robustness and Embedding Subband

The relationship between watermark robustness and embedded subbands watermark is evaluated in terms of normalized correlation NC and BER. To obtain the relationship between the different LH and LL subbands with embedding domain and watermark robustness to different attack, we studied the same AD.avi

Table 10: Performance Measurements in Terms of Computational Time.

Method	Algorithm	Scene Change Detector	Embedding Time	Extraction Using DWT	Total Computational Time
Embedding in all frames	Algorithm 1 (LH subband)	–	100.3154 s	5.50359 s	105.8190 s
Embedding in scene-changed frame		20.6235 s	17.6465 s	4.07526 s	21.7215 s
Embedding in all frames	Algorithm 2 (LL subband)	–	99.3155 s	5.50359 s	145.3220 s
Embedding in scene-changed frame		20.6235 s	16.6263 s	3.94603 s	20.5723 s

Table 11: Embedding and Extraction Time per Frame.

Algorithm	Embedding Time Per Frame	Extraction Time Per Frame
Algorithm 1 (watermarking with LH subband)	0.5514 s	0.12735 s
Algorithm 2 (watermarking with LL subband)	0.5195 s	0.12300 s

Figure 7: (A–D) Comparative Analysis of Robustness against Different Types of Attack between the LL Subband and the LH Subband Based on Normalized Correlation. (A) Image Processing Attack; (B) Geometrical Attacks; (C) Video Attacks; (D) JPEG Compression.

watermarked video sequence together with noise contamination or attacks with different subbands applied. Table 4 shows the NC values when the watermark is embedded into each one of the LH or LL wavelet subbands with the same embedding factor of 0.05 listed in different tables and depicted in Figure 7A–D.

The comparative performance analysis of Algorithms 1 and 2, i.e. watermarking using the LL and LH subbands using the parameters NC, SSIM, and BER with and without different attacks, is presented in Tables 4 and 12. The parameter values of the different tables using different attacks show that the embedded watermark is not highly robust when it is embedded into the detail subbands, i.e. LH subband, although these subbands offer better watermark imperceptibility, which is confirmed with results showing the SSIM parameter. The experimental result shows that watermarking using LL subband performs better and gives a more robust system. Both the proposed systems are robust against image processing attacks like salt and pepper noise, speckle noise, Gaussian LPF, blurring, sharpening, motion blur, normal blur, geometrical attacks like rotation, resizing and stretching, JPEG compression attack, temporal attacks like frame averaging, frame dropping, and frame swapping [2, 9, 34]. Both the algorithms are not robust against Gaussian noise. The percentage improvement in the performance in terms of SSIM, BER, and NC with the LL subband is given in Table 13.

5 Graphical Representation

Graphical representations of scene-based watermarking using the LL subband in terms of NC, BER, and SSIM are shown in Figure 8A–C after different attacks.

6 Application

Other than copyright protection, it also has copy protection as another application. Both the proposed systems can prove the ownership by extracting the watermark, containing information about time, date, and location identification [15]. As the proposed method is resilient to geometrical distortions like rotation

Figure 8: Analysis of Different Attacks in Watermarking Using the LL Subband (Algorithm 2). (A) Robustness in Terms of NC; (B) Robustness in Terms of BER; (C) Imperceptibility Analysis in Terms of Similarity Index After Different Attacks.

Table 12: Comparative analysis of SSIM between Watermarking Using LL and LH Subbands.

Attack Category	S. No.	Attack Subband Used	SSIM	
			LL Subband (Algorithm 2)	LH Subband (Algorithm 1)
Without attack	1	Without attack	1.0000000	1.0000000
Image processing attack	2	Salt and pepper noise	0.7271290	0.72959350
	3	Gaussian noise	0.3320910	0.35477024
	4	Speckle noise	0.7305794	0.70507169
	5	Gaussian LPF	0.9973051	0.98151565
	6	Blurring	0.9183954	0.78572745
	7	Sharpening	0.9839625	0.92820345
	8	Normal blur	0.7541251	0.63042296
	9	Motion blur	0.7812913	0.65706859
	10	Rotation	0.8845209	0.74551231
Geometrical attack	11	Resizing	1.0000000	1.00000000
	12	Stretching	1.0000000	1.00000000
	13	JPEG compression	0.7234546	0.7440630
Video attack	14	Frame averaging	0.7648609	0.65819758
	15	Frame dropping	0.9038462	0.90384615
	16	Frame swapping	0.7997019	0.71307598

and resizing depicted in Tables 5 and 6, JPEG compression attack at different quality factors represented in Table 7, and frame dropping at different drop rates, the extracted watermark image identifies the time and location of piracy [11]. It helps trace the area from where piracy was done. Finally, the number of piracy suspects may be restricted.

Table 13: Improvement in the Result with the LL Subband.

Attack	% Reduction BER	% Improvement NC	% Improvement SSIM
Salt and pepper noise	01.54%	01.21%	00.33%
Gaussian noise	00.67%	02.90%	06.39%
Speckle noise	02.01%	09.40%	00.01%
Gaussian LPF	04.25%	10.25%	01.58%
Blurring	45.22%	19.67%	16.88%
Sharpening	50.65%	01.24%	06.00%
Normal blur	18.21%	18.55%	19.62%
Motion blur	22.01%	06.21%	18.90%
Rotation	27.25%	14.35%	18.64%
Resizing	00.00%	00.87%	00.00%
Stretching	00.00%	10.64%	00.00%
JPEG compression	00.51%	11.38%	23.33%
Frame averaging	01.15%	16.93%	16.20%
Frame dropping	00.00%	12.64%	00.00%
Frame swapping	07.02%	—	12.14%

Table 14: Performance Analysis of System Using Different Parameters (LL Subband).

Video and watermark image	MSE	PSNR	NC	SSIM	BER
AD.avi and watermark image	0.00367	72.48941	0.91779	1.0000	0.0000
Children.avi and watermark image	0.00117	77.46000	0.93428	1.0000	0.0000
Documentary.avi and watermark image	0.00778	69.22308	0.93428	1.0000	0.0000

Figure 9: Cover Video Frame, Watermark Image, and Extracted Watermark Image. (A) Cover Video; (B) Watermark Image; (C) Extracted Image.

To further prove it, an experiment was performed using three videos, Ad.avi, Children.avi, and Documentary.avi, with a watermark image containing the time, date, and identification where the LL subband is used for embedding. Table 14 proves the transparency using the parameters MSE, PSNR, and SSIM, and the robustness using the parameters NC and BER. Children.avi video with watermark image shows a maximum PSNR value of 77.46 dB and NC value of 0.93428, given in Table 14. Figure 9 shows the original and extracted image with 0.91779 NC value. The empirical result proves that the watermark is imperceptible, and it can be recovered even with geometrical distortions and other distortions. The proposed algorithms are helpful in identifying the location and time of piracy; however, the exact position cannot be identified.

Table 15: Comparison between the Proposed Algorithms and Previous Work (Based on Attack).

Attacks Used	Already Existing Schemes						Proposed Schemes	
	Thanh et al. [34]	Ahuja and Bedi [2]	Masoumi and Amiri [19]	Agilandeewari and Ganesan [1]	Ghosh et al. [12]	Chandrakar and Qureshi [7]	LL Subband (Algorithm 1)	LH Subband (Algorithm 2)
None	✓	✓	✓	✓	✓	–	✓	✓
Salt and pepper noise	✓	✓	✓	✓	✓	–	✓	✓
Gaussian noise	✓	✓	✓	✓	✓	–	✓	✓
Speckle noise	–	–	–	–	–	–	✓	✓
Gaussian LPF	–	–	–	–	–	–	✓	✓
Rotation	–	–	–	–	–	–	✓	✓
Blurring	–	–	–	–	–	–	✓	✓
Sharpening	–	–	–	–	–	–	✓	✓
Resizing	–	–	–	–	–	–	✓	✓
JPEG compression	–	–	–	–	–	–	✓	✓
Normal blur	–	–	–	–	–	–	✓	✓
Motion blur	–	–	–	–	–	–	✓	✓
Stretching	–	–	–	–	–	–	✓	✓
Frame averaging	✓	✓	✓	✓	–	–	✓	✓
Frame dropping	✓	✓	✓	✓	–	–	✓	✓
Frame swapping	✓	✓	✓	✓	–	–	✓	✓
Multiple attacks	–	–	–	–	–	–	✓	✓
Total attacks	5	5	5	5	2	2	16	16

Table 16: Comparative Analysis Based on Parameters.

Method	Technique Used	PSNR	MSE	NC	SSIM	BER	No. of Attacks
[18]	Histogram difference method (discrete multiwavelet domain)	50.6419	–	33.606	–	–	Withstand seven attacks – salt and pepper noise, Gaussian noise, speckle noise, Poisson's noise, Wiener filter, cropping Attack analysis not performed
[7]	Blue channel, DWT, and SVD (random frame selection)	39.7596	1.2114	0.9888	–	–	Attack analysis not performed
[31]	DWT (embedding in each frame) (LL subband)	55.0110	0.2052	0.8780	–	–	No attack analysis
[30]	Scaled wavelet transform technique with SVD and DCT	–	–	0.9500	–	–	Withstand on three attacks – cropping, mean attack, rotation attack
Proposed method	LH subband (histogram difference)	53.2163	0.3101	0.9247	1	0	Fifteen different types of attacks like image processing attacks geometrical, video, noise attack, filtering
	LL subband (histogram difference)	64.6727	0.0221	0.8136	1	0	

7 Comparison with Other Previously Reported Algorithms

The proposed algorithms' performance was compared with other previously reported schemes [1, 2, 7, 12, 19, 34] in Table 15. It is clear from the results that the proposed algorithms worked better than the other algorithms in surviving various attacks. Table 16 also presents the comparison with other algorithms in terms of

different parameters. It shows that both the proposed algorithms give better imperceptibility and that the proposed scheme functions more perfectly than the other existing schemes, except for Gaussian noise.

8 Conclusion and Outlook

This paper demonstrates the design and implementation of two watermarking techniques, both in the frequency domain, based on the DWT. The watermark image is embedded into the detail subband (LH subband) in the first scheme, and the approximate subband (LL subband) of the cover video frame is used in the second scheme. To reduce the processing time, the watermark is embedded only in those frames when a scene transition happens. For detecting scene transitions, SESAME is used as a method and, further, histogram differences of successive frames are chosen, as measures.

The imperceptibility of two proposed schemes in terms of PSNR and SSIM has been presented. The watermark can be clearly extracted from the scene-changed frame of the watermarked video stream. The experimental result shows that the system is robust against image processing attacks, geometrical attacks, JPEG compression, and video attacks, and also sustains multiple attacks. Robustness in terms of NC and BER is presented. Channel capacity is also calculated. The empirical results of comparative analysis of the two proposed algorithms prove that watermarking using the LL subband performs better. The proposed algorithms facilitate the protection of the copyright of videos, and are also helpful in tracing the time and location of piracy.

The results of both the proposed methods suggest that scene-based video watermarking with histogram-based successive estimation offers simplicity, flexibility, reduced computational time, better imperceptibility, and improved robustness than other similar digital video watermarking schemes.

The outcomes of this research can be improved in several ways: (i) the performance of the developed system can enhance the result with a higher level of decomposition; (ii) by adapting the algorithm for all the real-time formats of videos; and (iii) by estimating the exact position of the pirate.

Bibliography

- [1] L. Agilandeewari and K. Ganesan, A robust color video watermarking scheme based on hybrid embedding techniques, *Multimed. Tools Appl.* **75** (2016), 8745–8780.
- [2] R. Ahuja and S. S. Bedi, Copyright protection using blind video watermarking algorithm based on mpeg-2 structure, In: *International Conference on Computing, Communication and Automation (ICCCA2015)*, Greater Noida, UP, India, 15–16 May, pp. 1048–1053, IEEE, 2015.
- [3] M. Ali, C. W. Ahn and M. Pant, A robust image watermarking technique using SVD and differential evolution in DCT domain, *Opt. Int. J. Light Electron Optics* **125** (2014), 428–434.
- [4] M. Barni, F. Bartolini, R. Caldelli, A. D. Rosa and A. Piva, A robust watermarking approach for raw video, In: *Proc. 10th International Packet Video Workshop, PV2000*, 2000.
- [5] P. Campisi, D. Kundur and A. Neri, Robust digital watermarking in the ridgelet domain, *IEEE Signal Process. Lett.* **11** (2004), 826–830.
- [6] M. U. Celik, G. Sharma, A. M. Tekalp and E. Saber, Lossless generalized-LSB data embedding, *IEEE Trans. Image Process.* **14** (2005), 253–266.
- [7] P. Chandrakar and S. G. Qureshi, A DWT based video watermarking using random frame selection, *Int. J. Res. Advent Technol.* **3** (2015), 39–44.
- [8] I. Cox, M. Miller, J. Bloom, J. Fridrich and T. Kalker, *Digital Watermarking and Steganography*, Morgan Kaufmann, 2007.
- [9] C. Cruz-Ramos, R. Reyes-Reyes, M. Nakano-Miyatake and H. Pérez-Meana, A blind video watermarking scheme robust to frame attacks combined with MPEG2 compression, *J. Appl. Res. Technol.* **8** (2010), 323–337.
- [10] N. V. Dharwadkar and B. B. Amberker, Secure watermarking scheme for color image using intensity of pixel and LSB substitution, arXiv preprint arXiv:0912.3923 (2009). Available: <http://arxiv.org/ftp/arxiv/papers/0912/0912.3923.pdf>, Accessed November 2010.
- [11] N. K. Dubey and S. Kumar, A review of watermarking application in digital cinema for piracy deterrence, In: *2014 Fourth International Conference on Communication Systems and Network Technologies*, Bhopal, India, 7–9 April, pp. 626–230, 2014.

- [12] P. Ghosh, R. Ghosh, S. Sinha, U. Mukhopadhyay, D. K. Kole and A. Chakroborty, A novel digital watermarking technique for video copyright protection, In: *International Conference of Advanced Computer Science & Information Technology*, Pune, Maharashtra, India, 2012. DOI: 10.5121/csit.2012.2360.
- [13] S. Haohao, Contourlet based adaptive watermarking for color images, *IEICE Trans. Inf. Syst.* **92** (2009), 2171–2174.
- [14] I. G. Karyali and K. Berberidis, Efficient spatial image watermarking via new perceptual masking and blind detection schemes, *IEEE Trans. Inform. Forens. Secur.* **1** (2006), 256–274. Available: <http://ieeexplore.ieee.org/stamp.jsp?arnumber=01634366>, Accessed October 2010.
- [15] H. Kelkoul and Y. Zaz, Digital cinema watermarking state of art and comparison, *World Acad. Sci. Eng. Technol. Int. J. Comput. Elect. Autom. Control Inform. Eng.* **11** (2017), 2017.
- [16] D. Kundur and D. Hatzinakos, A robust digital image watermarking method using wavelet-based fusion, In: *Proceedings of the International Conference on Image Processing 1997*, California, USA, October 26–29, Vol. 1, pp. 544–547, IEEE, October 1997. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=andarnumber=647970>, Accessed November 2010.
- [17] G. Langelaar, I. Setyawan and R. Lagendijk, Watermarking digital image and video data, *IEEE Signal Process. Mag.* **17** (2000), 20–43.
- [18] N. Leelavathy, E. V. Prasad and S. S. Kumar, A scene based video watermarking in discrete multiwavelet domain, *Int. J. Multidiscipl. Sci. Eng.* **3** (2012) 12–16.
- [19] M. Masoumi and S. Amiri, A blind scene-based watermarking for video copyright protection, *AEU – Int. J. Electron. Commun.* **67** (2013), 528–535.
- [20] D. P. Mukherjee, S. Maitra and S. T. Acton, Spatial domain digital watermarking of multimedia objects for buyer authentication, *IEEE Trans. Multimed.* **6** (2004), 1–15. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=andarnumber=1261883>, Accessed August 2011.
- [21] A. Nikolaidis and I. Pitas, Asymptotically optimal detection for additive watermarking in the DCT and DWT domains, *IEEE Trans. Image Processing* **12** (2003), 563–571.
- [22] X. Niu and S. Sun, A new wavelet based digital watermarking for video, In: *Proc. IEEE Digital Signal Processing Workshop*, 2000.
- [23] M. Ouhsain and A. B. Hamza, Image watermarking scheme using nonnegative matrix factorization and wavelet transform, *Expert Syst. Appl.* **36** (2009), 2123–2129.
- [24] R. Pandey, A. K. Singh and B. Kumar, Iris based secure NROI multiple eye image watermarking for teleophthalmology, *Multimed. Tools Appl.* **75** (2015), 14381–14397.
- [25] A. Phadikar, S. P. Maity and H. Rahaman, Region specific spatial domain image watermarking scheme, In: *Advance Computing Conference, 2009*, IACC, USA, March 2009, pp. 888–893, IEEE International, 2009. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=04809133>, Accessed July 2015.
- [26] S. Ranjbar, F. Zargari and M. Ghanbari, A highly robust two-stage contourlet-based digital image watermarking method, *Signal Process. Image Commun.* **28** (2013), 1526–1536.
- [27] S. Roy and A. K. Pal, A robust blind hybrid image watermarking scheme in the RDWT-DCT domain using Arnold scrambling, *Multimed. Tools Appl.* **76** (2016), 1–40.
- [28] C. Serdean, M. Ambroze, M. Tomlinson and G. Wade, Combating geometrical attacks in a DWT based blind video watermarking system, In: *Proc. EurasiP – IEEE VIPPromCom*, pp. 263–266, 2002.
- [29] A. Sharma, A. K. Singh, and S. P. Ghrera, Robust and secure multiple watermarking technique for medical images, *Wireless Pers. Commun.* **92** (2017), 1611–1624.
- [30] D. K. Shaveta, Attack resistant robust video watermarking using scaled wavelet transform with SVD-DCT techniques, *Int. J. Modern Comput. Sci.* **3** (2015). Available: <https://www.ijeat.org/v6i3.php>.
- [31] D. Shukla and M. Sharma, Performance evaluation of video watermarking system using discrete wavelet transform for four subband, In: *International Conference on Cyber Security (ICCS) 2016*, held at Rajasthan Technical University, Kota, August 13–14, 2016.
- [32] A. K. Singh, Improved hybrid algorithm for robust and imperceptible multiple watermarking using digital images, *Multimed. Tools Appl.* (2016), DOI: 10.1007/s11042-016-3514-z.
- [33] K. Su, D. Kundur and D. Hatzinakos, A novel approach to collusion-resistant video watermarking, In: *SPIE Proc. 4675, Security and Watermarking of Multimedia Content IV*, pp. 491–502, 2002.
- [34] T. M. Thanh, P. T. Hiep, T. M. Tam and K. Tanaka, Robust semi-blind video watermarking based on frame-patch matching, *AEU – Int. J. Electron. Commun.* **68** (2014), 1007–1015.
- [35] R. B. Wolfgang, C. I. Podilchuk and E. J. Delp, Perceptual watermarks for digital images and video, *Proc. IEEE* **87** (1999), 1108–1126.
- [36] C. Wu, W. P. Zhu and M. N. Swamy, A watermark embedding scheme in wavelet transform domain, In: *TENCON, IEEE Region 10 Conference Proceedings A*, Chiang Mai, 21–24 November, pp. 279–282, 2004.
- [37] A. Zear, A. K. Singh and P. Kumar, A proposed secure multiple watermarking technique based on DWT, DCT and SVD for application in medicine, *Multimed. Tools Appl.* (2016), 1–20. DOI: 10.1007/s11042-016-3862-8.
- [38] A. Zear, A. K. Singh and P. Kumar, Multiple watermarking for healthcare applications, *J. Intell. Syst.* **27** (2018), 5–18.