SIViP (2010) 4:233–245
DOI 10.1007/s11760-009-0114-7

ORIGINAL PAPER

Video watermarking using wavelet transform and tensor algebra

Emad E. Abdallah · A. Ben Hamza · Prabir Bhattacharya

Received: 13 April 2008 / Revised: 2 April 2009 / Accepted: 7 April 2009 / Published online: 24 April 2009

Abstract We present a robust, hybrid non-blind MPEG video watermarking technique based on a high-order tensor singular value decomposition and the discrete wavelet transform (DWT). The core idea behind our proposed technique is to use the scene change analysis to embed the watermark repeatedly into the singular values of high-order tensors computed form the DWT coefficients of selected frames of each scene. Experimental results on video sequences are presented to illustrate the effectiveness of the proposed approach in terms of perceptual invisibility and robustness against attacks.

Keywords Video watermarking · Tensor singular value decomposition · Wavelet transform

1 Introduction

The use of digital video applications such as video-conferencing, digital television, digital cinema, distance learning, videophone, and video-on-demand has grown very rapidly over the last few years. Today it is much easier for the digital data owners to transfer multimedia data over the internet, and hence the data could be perfectly duplicated and rapidly redistributed on a large scale. Thus, the importance of copyright protection for multimedia data has became more critical. Digital watermarking is an effective way to protect the copyright of multimedia data even after its transmission [1]. Watermarking refers to the process of adding a hidden structure, called a watermark, into a multimedia data that carries either, the information about the owner of the cover or, the recipient of the original data object. Watermark applications include broadcast monitoring, copy control, transaction tracing, and copyright protection. Robustness, invisibility and security are the three most important properties that need to be satisfied for such applications [1,2].

In image watermarking, the degradation of the watermarked image should not be perceptible to the human observer [1–3]. A variety of watermarking techniques have been proposed to embed a robust watermark into digital images. These techniques can be divided into two main categories according to the embedding domain of the cover image: the spatial domain methods and the transform domain methods. The spatial domain methods are the earliest and simplest watermarking techniques but they have a low information hiding capacity, and the watermark can be easily erased by lossy image compression. On the other hand, the transform domain approaches insert the watermark into the transform coefficients of the original image "cover", yielding more information embedding and more robustness against watermarking attacks. Recent popular transforms include the discrete cosine transform (DCT) [4], the discrete wavelet transform (DWT) [5], and the discrete Fourier transform (DFT). Image watermarking techniques can be extended easily to watermark video image sequences [6,7]. However, video watermarking schemes need to meet some other challenges such as the large volume of inherently redundant data between frames, unbalance between motion and motionless regions [8], and the real-time requirements in video broadcasting that make the video signals highly susceptible to pirate attacks including frame averaging, frame dropping, frame swapping, and statistical analysis [9].

The early video watermarking techniques add a visible signature or a logo to the video frames [1]. These watermarks do not usually cover significant areas of the video frames, making them easy to remove by a cropping attack.

E. E. Abdallah · A. Ben Hamza (✉) · P. Bhattacharya
Concordia Institute for Information Systems Engineering,
Concordia University, Montreal, Canada
e-mail: hamza@ciise.concordia.ca

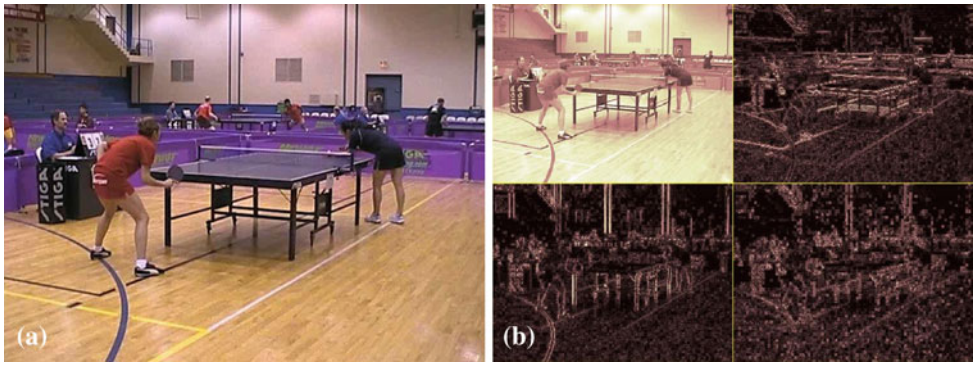

Fig. 1 One frame of the tennis video sequence with one level of DWT decomposition

Recently, a real-time digital video watermarking scheme [10] has been proposed to embed the watermark in intra-pictures of an MPEG video sequence by modifying the variable length codes (VLCs) directly in order to avoid inverse quantization. The main advantage of modifying the VLCs is that the perceptual degradation of video quality caused by the embedded watermark is minimized [10]. In Ref. [11], a blind MPEG2 video watermarking technique was proposed by focusing on geometric attacks. The DFT of 3D chunks of a video scene was used in Ref. [12] for video watermarking, where the embedding and the extraction algorithms are applied to uncompressed video data. In Ref. [13], only the DCT coefficients of the intra-frames in the MPEG compressed video are watermarked, and the spread spectrum signal was used as a copyright information that was added to the non-zero DCT coefficients under the condition of not increasing the bit rate. Embedding the watermark in the uncompressed domain was proposed in Ref. [14], where the watermark was embedded in the intra-frames by adopting the block matching algorithm to find the motion vector of each block and also by using the motion feature to embed the watermark.

Motivated by the good performance of the 2D image watermarking techniques proposed in Ref. [15,16], we present in this paper a scene change watermarking approach using a hybrid scheme based on DWT and tensor singular value decomposition (TSVD). Our approach generalizes the method proposed in Ref. [16] by embedding the watermark data in all the frequencies of the video scenes. The key idea is to apply the TSVD to the four wavelets sub-bands of the video frames viewed as a 3D tensor with two dimensions in space and one dimension in time. The algorithm is based on our previous algorithm [17] that employs the higher order tensors. The new algorithm is non-blind and consequently it requires the original video sequence in the extraction stage. The experimental results show that the proposed scheme is robust against a variety of attacks including frame dropping, frame averaging, frame swapping, geometric transformations, adaptive random noise, low pass filtering, and histogram equalization.

The rest of the paper is organized as follows: in Sect. 2, we briefly review the DWT and the multidimensional tensor singular value decomposition, in Sect. 3, we provide a brief review of some previous works that are closely related to our proposed watermarking scheme. In Sect. 4, we introduce the proposed approach and describe in detail the watermark embedding and extraction algorithms. Experimental results are presented in Sect. 5 to demonstrate the performance of the proposed watermarking scheme in comparison with existing methods. Finally, we conclude in Sect. 6.

2 Background

2.1 Discrete wavelet transform

The DWT has been used successfully in many image processing applications including noise reduction, edge detection, and compression [16,18]. The DWT is computed by successive low-pass and high-pass filtering of the discrete time-domain signal. Its significance is in the manner it connects the continuous-time multiresolution to discrete-time filters. At each level, the high pass filter produces detailed information, while the low pass filter associated with scaling function produces coarse approximations. We use a 2D version of the analysis and synthesis filter banks by applying a 1D analysis filter bank to the columns of the image and then to the rows. If the image has m rows and n columns, then after applying the 2D analysis filter bank we obtain four sub-band images (LL, LH, HL, and HH), each having $m/2$ rows and $n/2$ columns.

Figure 1a, b shows an example of one frame of the tennis video sequence with one level DWT. Figure 2 shows the wavelet coefficients of all the four sub-bands shown in Fig. 1b, where it can be seen that the wavelet coefficients of the LL sub-band are the highest among all the coefficients of the other sub-bands.

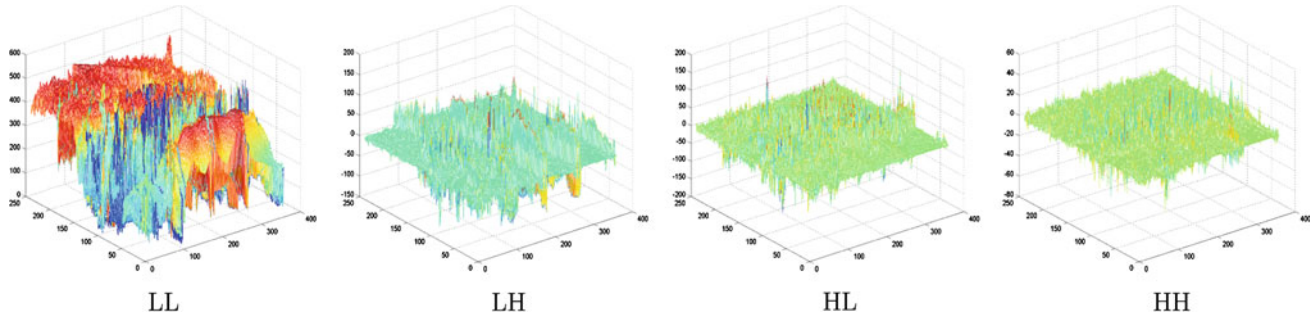

Fig. 2 DWT coefficients of all the four sub-bands shown in Fig. 1b

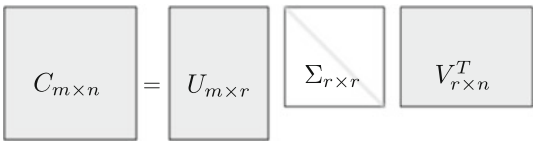

Fig. 3 Illustration of the SVD approximation

2.2 Tensor algebra

2.2.1 Singular value decomposition of 2D array

The SVD of a 2D image $\boldsymbol{C}$ of size $m \times n$ is given by $\boldsymbol{C} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, where $\boldsymbol{U}$ is an orthogonal matrix ($\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$), $\boldsymbol{\Sigma} = \text{diag}\,(\lambda_i)$ is a diagonal matrix of singular values λ_i, $1 \leq i \leq r$, arranged in decreasing order, and $\boldsymbol{V}$ is an orthogonal matrix ($\boldsymbol{V}^T\boldsymbol{V} = \boldsymbol{I}$) as depicted in Fig. 3. The columns of $\boldsymbol{U}$ are the left singular vectors, whereas the columns of $\boldsymbol{V}$ are the right singular vectors of the image $\boldsymbol{C}$.

2.2.2 Multidimensional tensor singular value decomposition

Higher order singular value decomposition (HOSVD) has been proposed in Ref. [19] to analyze multilinear structures. Transforming a 3D tensor into a matrix is usually referred to as a "matricization" process [20–22]. The n-mode matricizing of a tensor $\mathbf{A} \in \mathbb{R}^{I_1\times I_2\cdots I_N}$ is denoted by a matrix $\mathbf{A}_n \in \mathbb{R}^{I_n\times(I_{n+1}\times\cdots\times I_N\times I_1\times\cdots\times I_{n-1})}$, as is shown in Fig. 4.

The n-mode product of a tensor $\mathbf{A}$ by a matrix $\mathbf{U} \in \mathbb{R}^{J_n\times I_n}$ is an $I_1 \times \cdots \times I_{n-1} \times J_n \times I_{n+1} \times \cdots \times I_N$ tensor denoted by $\mathbf{A} \times_n \mathbf{U}$, whose entries are defined by:

$$[\mathbf{A} \times_n \mathbf{U}]_{i_1 i_2 \ldots i_{n-1} j_n i_{n+1} \ldots i_N} = \sum_{i_n} a_{i_1 i_2 \ldots i_{n-1} i_n i_{n+1} \ldots i_N} u_{j_n i_n} \tag{1}$$

where $a_{i_1 i_2 \ldots i_{n-1} i_n i_{n+1} \ldots i_N}$ is an entry of $\mathbf{A}$, and $u_{j_n i_n}$ is an entry of $\mathbf{U}$.

The n-mode product $\times_n$ satisfies commutability [19,20]. Given a tensor $\mathbf{A} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_m\times\cdots\times I_n\times\cdots\times I_N}$ and two matrices $\mathbf{B} \in \mathbb{R}^{J_n\times I_n}$ and $\mathbf{C} \in \mathbb{R}^{J_m\times I_m}$, we have

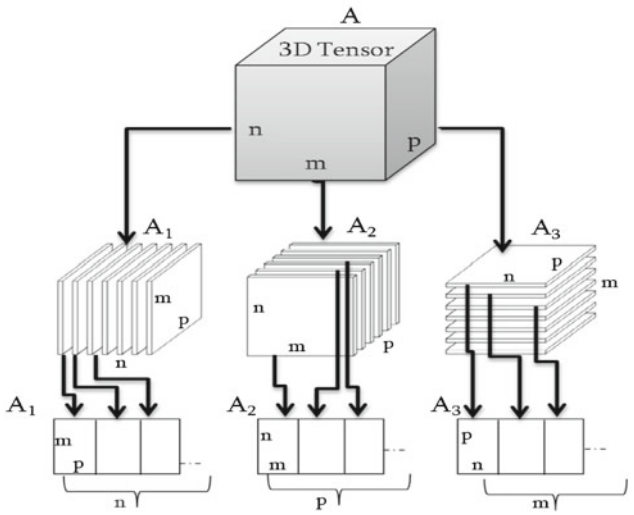

Fig. 4 Illustration of matricizing a third-order tensor $\mathbf{A}$ into a matrix in three ways. $\mathbf{A_1} \in \mathbb{R}^{n\times(m\times p)}$ is the one-mode matricizing of the tensor $\mathbf{A}$. $\mathbf{A_2} \in \mathbb{R}^{p\times(n\times m)}$ is the two-mode matricizing of the tensor $\mathbf{A}$. $\mathbf{A_3} \in \mathbb{R}^{m\times(p\times n)}$ is the three-mode matricizing of the tensor $\mathbf{A}$

$$\mathbf{A} \times_n \mathbf{B} \times_m \mathbf{C} = \mathbf{A} \times_m \mathbf{C} \times_n \mathbf{B} \tag{2}$$

In this paper, we deal mainly with video sequences that are represented as a 3D tensor with two dimensions in space and one dimension in time. Let $\boldsymbol{A}$ be 3D video tensor of size $m \times n \times p$. The tensor $\boldsymbol{A}$ can be rearranged into a matrix of size $k \times \ell$ in three different ways: left-right matrix $\boldsymbol{A_1}$, front-back matrix $\boldsymbol{A_2}$, and top-bottom matrix $\boldsymbol{A_3}$, as shown in Fig. 4. Clearly, the number of elements in the matrices $\boldsymbol{A_1}$, $\boldsymbol{A_2}$ and $\boldsymbol{A_3}$ must be the same as the number of elements in the tensor $\boldsymbol{A}$.

Extending matrix decompositions such as the SVD to higher-order tensors has proven to be quite difficult [21]. Given an $m \times n \times p$ tensor $\boldsymbol{A}$, the Tucker decomposition (see Fig. 5) is given by:

$$\begin{aligned}\boldsymbol{A} &= \boldsymbol{\Sigma} \times_1 \boldsymbol{U} \times_2 \boldsymbol{V} \times_3 \boldsymbol{W} \\ &= \sum_{i=1}^{r_1}\sum_{j=1}^{r_2}\sum_{k=1}^{r_3}\sigma_{ijk}(u_i \otimes v_j \otimes w_k)\end{aligned} \tag{3}$$

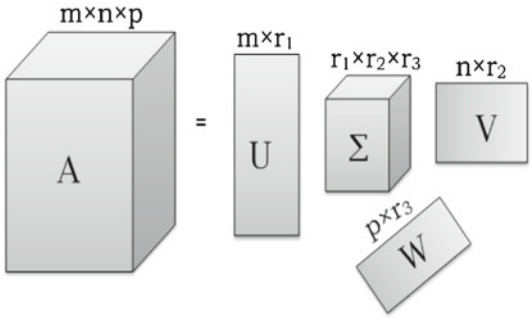

Fig. 5 Tucker decomposition of a 3D tensor

where $r_1 \leq m$, $r_2 \leq n$, $r_3 \leq p$ and the columns of $\boldsymbol{U}$, $\boldsymbol{V}$, and $\boldsymbol{W}$ are the left singular vectors of the matrices $\boldsymbol{A_1}$, $\boldsymbol{A_2}$ and $\boldsymbol{A_3}$. The tensor $\boldsymbol{\Sigma} = (\sigma_{ijk})$, is called the core tensor and it is given by:

$$\boldsymbol{\Sigma} = \boldsymbol{A} \times_1 \boldsymbol{U}^T \times_2 \boldsymbol{V}^T \times_3 \boldsymbol{W}^T \tag{4}$$

The core tensor dose not necessarily have the same dimension as $\boldsymbol{A}$. In general, we can have either orthogonal columns of $\boldsymbol{U}$, $\boldsymbol{V}$, and $\boldsymbol{W}$ or a diagonal core tensor $\boldsymbol{\Sigma}$ [19].

Applying SVD to the matrices $\boldsymbol{A_1}$, $\boldsymbol{A_2}$ and $\boldsymbol{A_3}$ yields:

$$\begin{aligned} \boldsymbol{A}_1 &= \boldsymbol{U}\boldsymbol{D}_1\boldsymbol{G}_1^T \\ \boldsymbol{A}_2 &= \boldsymbol{V}\boldsymbol{D}_2\boldsymbol{G}_2^T \\ \boldsymbol{A}_3 &= \boldsymbol{W}\boldsymbol{D}_3\boldsymbol{G}_3^T \end{aligned} \tag{5}$$

where the columns of $\boldsymbol{G}_1$, $\boldsymbol{G}_2$, and $\boldsymbol{G}_3$ are the right singular vectors of $\boldsymbol{A_1}$, $\boldsymbol{A_2}$ and $\boldsymbol{A_3}$ respectively. Moreover, we have

$$\begin{aligned} \boldsymbol{A}_1 &= \boldsymbol{U}\boldsymbol{\Sigma}_1(\boldsymbol{V} \otimes \boldsymbol{W})^T \\ \boldsymbol{A}_2 &= \boldsymbol{V}\boldsymbol{\Sigma}_2(\boldsymbol{W} \otimes \boldsymbol{U})^T \\ \boldsymbol{A}_3 &= \boldsymbol{W}\boldsymbol{\Sigma}_3(\boldsymbol{U} \otimes \boldsymbol{V})^T \end{aligned} \tag{6}$$

where $\boldsymbol{\Sigma}_1 = \boldsymbol{D}_1\boldsymbol{G}_1^T(\boldsymbol{V} \otimes \boldsymbol{W})$, $\boldsymbol{\Sigma}_2 = \boldsymbol{D}_2\boldsymbol{G}_2^T(\boldsymbol{W} \otimes \boldsymbol{U})$ and $\boldsymbol{\Sigma}_3 = \boldsymbol{D}_3\boldsymbol{G}_3^T(\boldsymbol{U} \otimes \boldsymbol{V})$

3 Related works

In this section, we review three representative methods for digital video watermarking that are closely related to our proposed approach. We briefly discuss their mathematical foundations and algorithmic methodologies.

3.1 Non-blind video watermarking using tensor singular value decomposition

In Ref. [17], the singular values of the watermark image is embedded in the intra-frames of an MPEG video sequence. Let $\boldsymbol{V}$ be the cover video sequence, and $\boldsymbol{\Omega_m}$ be a watermark image of size $m \times m$. The video frames are divided into groups that are represented by 3D tensors. Then, the tensor is matricized to obtain the matrices $\boldsymbol{A_1}$, $\boldsymbol{A_2}$, and $\boldsymbol{A_3}$. The SVD is applied to the matrices $\boldsymbol{A_1}$, $\boldsymbol{A_2}$, and $\boldsymbol{A_3}$ to calculate the 3D singular values (SVs) tensor $\boldsymbol{\Sigma_{3D}} = \boldsymbol{A} \times_1 \boldsymbol{U}^T \times_2 \boldsymbol{V}^T \times_3 \boldsymbol{W}^T$, where $\boldsymbol{U}$, $\boldsymbol{V}$, and $\boldsymbol{W}$ are the left singular vectors of $\boldsymbol{A_1}$, $\boldsymbol{A_2}$, and $\boldsymbol{A_3}$ respectively. The SVD is then applied to $\boldsymbol{\Omega_m}$ to get its singular values (λ_ω^i), $1 \leq i \leq m$. The largest SVs of $\boldsymbol{\Sigma_{3D}}$ are modified with the SVs of $\boldsymbol{\Omega_m}$ using the additive model:

$$\widehat{\lambda^i} = \lambda^i + \alpha\lambda_\omega^i \tag{7}$$

where λ^i are the singular values of $\boldsymbol{\Sigma_{3D}}$, $\widehat{\lambda^i}$ denotes the distorted SVs, and α is a constant scaling factor referred to as the watermark strength. Finally, the watermarked video sequence is produced using the watermarked tensor: $\boldsymbol{A}_w = \widehat{\boldsymbol{\Sigma}}_{3D} \times_1 \boldsymbol{U} \times_2 \boldsymbol{V} \times_3 \boldsymbol{W}$, where $\widehat{\boldsymbol{\Sigma}}_{3D}$ denotes the modified 3D core tensor with the diagonal elements λ^i.

The algorithm is invertible, that is the watermark image can be extracted from the watermarked video sequence by extracting the singular values of the visual watermark using: $\widehat{\lambda}_w^i = (\widehat{\lambda^i} - \lambda^i)/\alpha$, where λ^i and $\widehat{\lambda^i}$ are the original and the watermarked singular values respectively. Therefore, the extracted watermark image is given by $\widehat{\boldsymbol{\Omega}} = \boldsymbol{U}_w\widehat{\boldsymbol{\Sigma}}_w\boldsymbol{V}_w^T$, where $\boldsymbol{U}_w$ and $\boldsymbol{V}_w$ are the left and right singular vectors of $\boldsymbol{\Omega_m}$ respectively, and $\widehat{\boldsymbol{\Sigma}}_w = \text{diag}(\widehat{\lambda}_w^i)$ is the extracted matrix of SVs of $\widehat{\Omega}$.

3.2 Non-blind MPEG video watermarking in the DWT domain

The idea of the method introduced in Ref. [23] is to embed a binary pattern in the form of a binary image as an invisible watermark in the four wavelet sub-bands of each intra-frame of the MPEG video. This could be done as follows. The cover video frame $\boldsymbol{C}$ is decomposed into four sub-bands: the approximation coefficient LL, and the detailed coefficients HL, LH, and HH. The DWT coefficients of each sub-band $C^k \in \{\text{LL,HL,LH,HH}\}$ are modified with the binary image as follows

$$\widehat{C}_{ij}^k = C_{ij}^k + \alpha_k w_{ij}, \quad 1 \leq k \leq 4 \tag{8}$$

where C_{ij}^k and $\widehat{C}_{ij}^k$ denote the original and the distorted DWT coefficients of the sub-band of the watermarked frame respectively, and w_{ij} denotes the (i, j)th pixel value of the watermark image. Hence, we get the four modified sub-bands. Then, the inverse DWT is applied using the four sets of the modified DWT coefficients to produce the watermarked frame.

The algorithm is invertible and the watermark can be extracted from the watermarked video frames by extracting the binary values of the visual watermark using: $w_{ij}^{*k} = (\widehat{C}_{ij}^k - C_{ij}^k)/\alpha_k$.

3.3 Blind hybrid scene-based watermarking scheme

In Ref. [24], the watermark is divided into small parts as a preprocess before embedding these parts into scenes. Four levels of DWT are applied to all frames in a video sequence, producing a low-frequency sub-band LL4, and three series of high-frequency sub-bands. Different watermarks are embedded in frames of different scenes and an identical watermark is used for each frame in the same scene. The watermark embedding is applied to the video frame by changing the position of some DWT coefficients. Let W_j the jth pixel value of a corresponding watermark image and C_i the ith DWT coefficient of the video frame then,

If $W_j = 1$, exchange C_i with the maximum of $(C_i, C_{i+1}, C_{i+2}, C_{i+3}, C_{i+4})$

Else exchange C_i with the minimum of $(C_i, C_{i+1}, C_{i+2}, C_{i+3}, C_{i+4})$,

This algorithm is blind, that is, the retrieval of the embedded watermark does not need the original video frames. To extract the watermark, each video frame is transformed to the wavelet domain with four levels. Then the watermark is extracted using the following condition:

If $\mathrm{WC}_i > \text{median}\ (\mathrm{WC}_i, \mathrm{WC}_{i+1}, \mathrm{WC}_{i+2}, \mathrm{WC}_{i+3}, \mathrm{WC}_{i+4})$ then $\mathrm{EW}_j = 1$

Else $\mathrm{WC}_i < \text{median}\ (\mathrm{WC}_i, \mathrm{WC}_{i+1}, \mathrm{WC}_{i+2}, \mathrm{WC}_{i+3}, \mathrm{WC}_{i+4})$ then $\mathrm{EW}_j = 0$

where WC_i is the ith DWT coefficient of the watermarked video frame, and EW_j is the jth pixel value of a extracted watermark image.

To improve the robustness against image processing attacks on video frames, a hybrid approach by watermarking the audio signals and using different watermarking schemes for different scenes was proposed in Ref. [24]. The watermark is still decomposed into different parts which are embedded in the corresponding frames of different scenes in the original video. Each part of the watermark, however, is embedded with a different watermarking scheme. Within a scene, all the video frames are watermarked with the same part of a watermark by the same watermarking scheme. Thus, the hybrid approach enhances the robustness against image processing attacks.

4 Proposed watermarking scheme

In Ref. [16], the authors showed experimentally that embedding the watermark in the low and high frequency components of an image increases the robustness against attacks. More specifically, embedding the watermark in low frequency components increases the robustness to the attacks that have low frequency characteristics such as filtering, lossy compression, and geometric distortions, whereas embedding the watermark in the middle and high frequency components is typically less robust against low pass filtering and small geometric deformations of the image, but it is more robust to noise addition, contrast adjustment, gamma correction, and histogram manipulation. Therefore, the goal of our proposed approach is to apply multiple transforms to selected frames of the cover video in order to embed the watermark many times in all the frequencies that provide better robustness against attacks. This would amplify the difficulty of destroying the watermark from all the frequencies, and provide a high visual quality of the watermarked video sequence.

In MPEG multiplexed stream (MPEG1 system and MPEG2 program stream), there are typically three kinds of coded images in each group of pictures: I (intra) frame compressed using only intraframe coding, P (predicted) frame coded with motion compression using past I-frames or P-frames, and B (bidirectional) frame coded by motion compensation by either past or future I or P frames, In order to achieve a low complexity and improve the robustness, we only used the I-frames to embed the watermark. The watermark Ω used for embedding is a sequence of random numbers that are produced from an integer random number generator. Embedding a watermark into each and every frame in the video using a signal image leads to problems of maintaining statistical and perceptual invisibility [8]. Therefore, an identical watermark has been used for each group of the I-frames in the same scene, and different watermarks for different scenes. We partition the MPEG video into scenes by counting the percentage of each type of blocks in a frame [11].

4.1 Watermark embedding

The watermark embedding process is described in Algorithm 1.

4.2 Watermark extraction

The watermark extraction is performed by applying the first six steps of the watermark embedding process to the original as well as the watermarked video sequences. Then, for each set of the I-frames we extract the watermark vector four times from the four tensors representing the transformed wavelet coefficients using: $\widehat{w}^i = (\widehat{\lambda^i} - \lambda^i)/\alpha$, where λ^i and $\widehat{\lambda^i}$ are the original and the watermarked singular values respectively. Finally we select the extracted watermark vector that has the highest correlation with the original watermark. Figure 7 shows an example of video watermarking using our proposed scheme. Clearly the difference between the original and the watermarked videos is unnoticeable to the human observer.

It is worth pointing out that the proposed method is computationally inexpensive due to the fact that the watermark

Algorithm 1 Watermark embedding algorithm

Input *Original Video and a random watermark vector* Ω_m.
Output*: Watermarked video.*
◇ The video is partitioned into scenes.
for each scene **do**
1- Convert the I-frames from RGB to YUV (Y represents the luminance component i.e. the brightness, U and V represent the chrominance components i.e color). In order to make the watermark imperceptible, we use the luminance layer to embed the watermark and we leave the chrominance layer unchanged.
2- Apply DWT to the converted luminance layers (Y) of the I-frames to obtain 4 sub-bands of each frame (LL, LH, HL, HH).
3- For each set of I-frames, divide the sub-bands into four chunks (groups). The first group is created from LL sub-bands, the second one from LH sub-bands, the third one from HL sub-bands, and the fourth one from HH sub-bands. All these groups are represented as 3D tensors as shown in Fig. 6.
4- Matricize the four tensors in three different ways (left-right, front-back, and top-bottom) to obtain $\boldsymbol{A}_1^k$, $\boldsymbol{A}_2^k$, $\boldsymbol{A}_3^k$ respectively, where the index $k \in \{1, 2, 3, 4\}$ represents the four tensors.
5- For each tensor, apply SVD to the matrices $\boldsymbol{A_1}$, $\boldsymbol{A_2}$, and $\boldsymbol{A_3}$ that is, $\boldsymbol{A}_1 = \boldsymbol{U}\boldsymbol{D}_1\boldsymbol{G}_1^T$, $\boldsymbol{A}_2 = \boldsymbol{V}\boldsymbol{D}_2\boldsymbol{G}_2^T$, and $\boldsymbol{A}_3 = \boldsymbol{W}\boldsymbol{D}_3\boldsymbol{G}_3^T$.
6- Calculate the 3D singular values (SVs) matrix $\boldsymbol{\Sigma_{3D}} = \boldsymbol{A} \times_1 \boldsymbol{U}^T \times_2 \boldsymbol{V}^T \times_3 \boldsymbol{W}^T$.
7- For all the tensors, modify the largest SVs of $\boldsymbol{\Sigma_{3D}}$ with a random watermark vector $\boldsymbol{\Omega_m}$ using: $\widehat{\lambda^i} = \lambda^i + \alpha w^i$, where $\widehat{\lambda^i}$, λ^i are the distorted and the original SVs of $\boldsymbol{\Sigma_{3D}}$ respectively, α is a constant scaling factor, and $1 \leq i \leq m$ where m is the size of the watermark vector.
8- Produce the watermarked tensor $\boldsymbol{A}_w = \widehat{\boldsymbol{\Sigma}}_{3D} \times_1 \boldsymbol{U} \times_2 \boldsymbol{V} \times_3 \boldsymbol{W}$, where $\widehat{\boldsymbol{\Sigma}}_{3D}$ is the modified 3D core tensor.
9- Use $\boldsymbol{A}_w$ to produce the modified luminance layer of the I-frames.
10- Finally use the modified I-frames to produce the watermarked cover video sequence.
end for

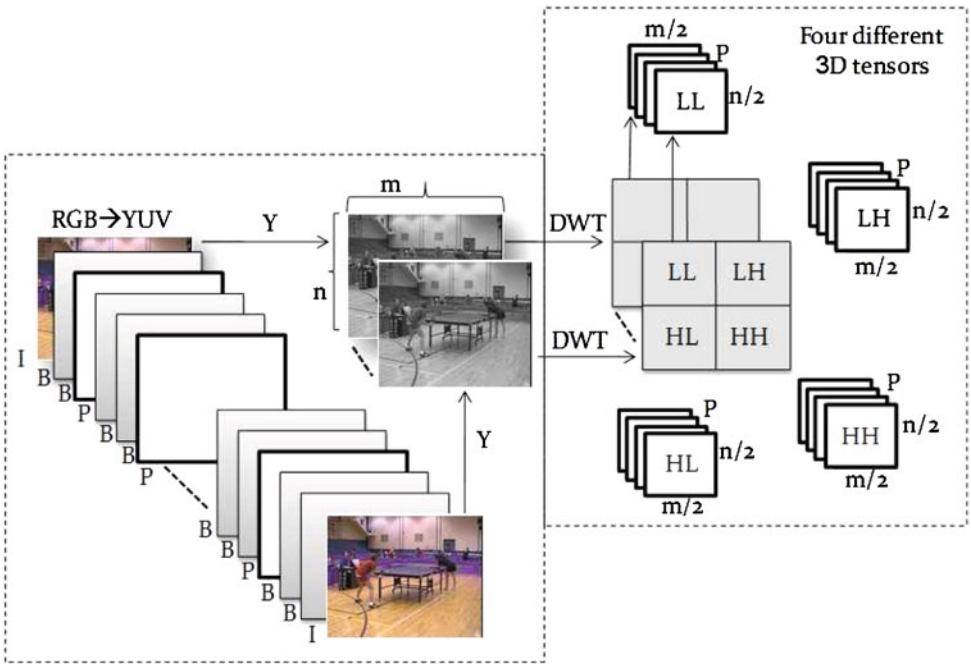

Fig. 6 Illustration of a multidimensional four tensors produced from one group of the I-frames

embedding and extraction algorithms are not applied to each and every single frame of the video image sequence. However, these algorithms are applied to 3D tensors that are computed from the I-frames of the cover video.

5 Experimental results

We tested the performance of the proposed watermarking scheme on several video sequences that are shown in Fig. 8. Sequences of 128 positive integers between 1 and 100 are randomly generated and used as watermarks in these experiments. One sequence of 128 integers is repeatedly embedded in the video tensors of the same scene, and different watermarks for different scenes. The experiments are performed to verify the watermark imperceptibility and robustness against attacks.

5.1 Watermark imperceptibility

In order to achieve a high visual quality of the watermarked video sequence, the watermark strength factor α should be taken into consideration. The strength factor should be small enough to keep the watermark imperceptible to the human

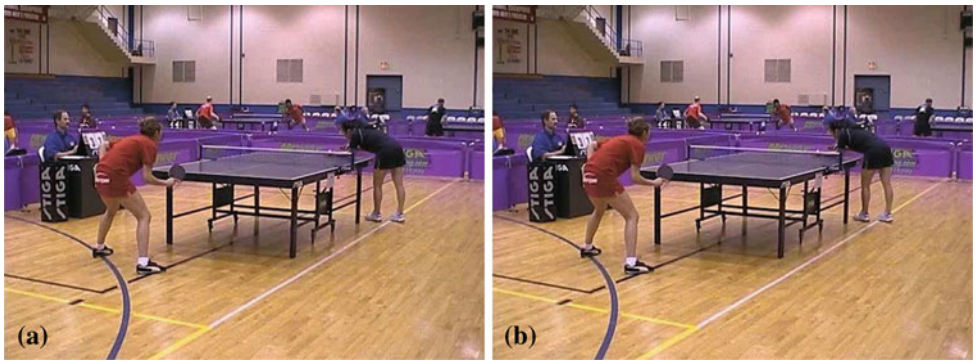

Fig. 7 **a** Original table tennis video frame, **b** Watermarked table tennis video frame

Fig. 8 Sample frames from videos used in the experiments

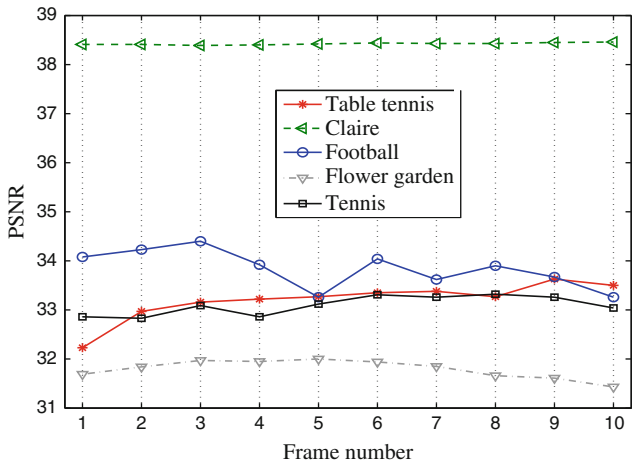

Fig. 9 PSNR results for different video sequences. Ten continues frames are chosen from one video scene

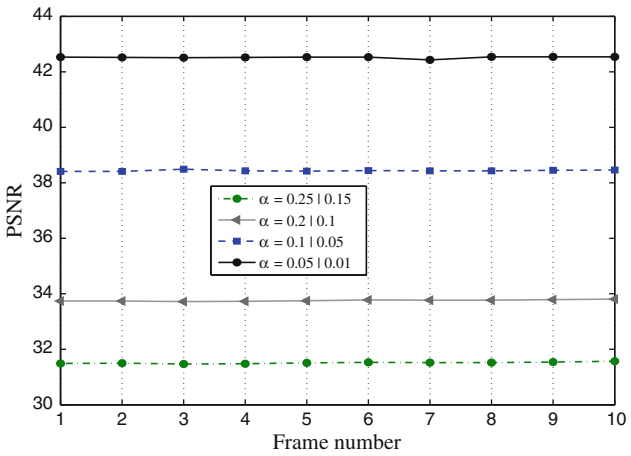

Fig. 10 PSNR for the Claire video sequence. PSNR between 10 frames and their corresponding watermarked frames with different strength factors. The *left-hand side* α value is used for the LL band and the *right-hand side* α value is used for the LH, HL and HH sub-bands

observer, and large enough to resist as many attacks as possible. Experimentally a constant scaling factor $\alpha = 0.1$ was used for the tensors representing the LL sub-bands and $\alpha = 0.05$ for all the other tensors. The strength factors are chosen according to the wavelet coefficients of the sub-bands frames. The LL sub-bands have the lowest frequency components of the cover video frames and the highest wavelet coefficients (highest magnitude). The (HL, LH and HH) sub-bands have very similar wavelet coefficients values, and therefore we used the same strength factor for all middle and high frequency sub-bands.

Table 1 Invisibility test using human observers

	Flower garden (%)	Claire (%)	Table tennis (%)	Football (%)
$\alpha = 0.1 \vert 0.05$	0	0	0	0
$\alpha = 0.2 \vert 0.1$	0	4	4	0
$\alpha = 0.2 \vert 0.3$	25	25	25	25

Percentage of number of people who were able to see a difference between the original video and the watermarked video under different strength factors. The left-hand side α value is used for the LL band and the right-hand side α value is used for the LH, HL and HH sub-bands

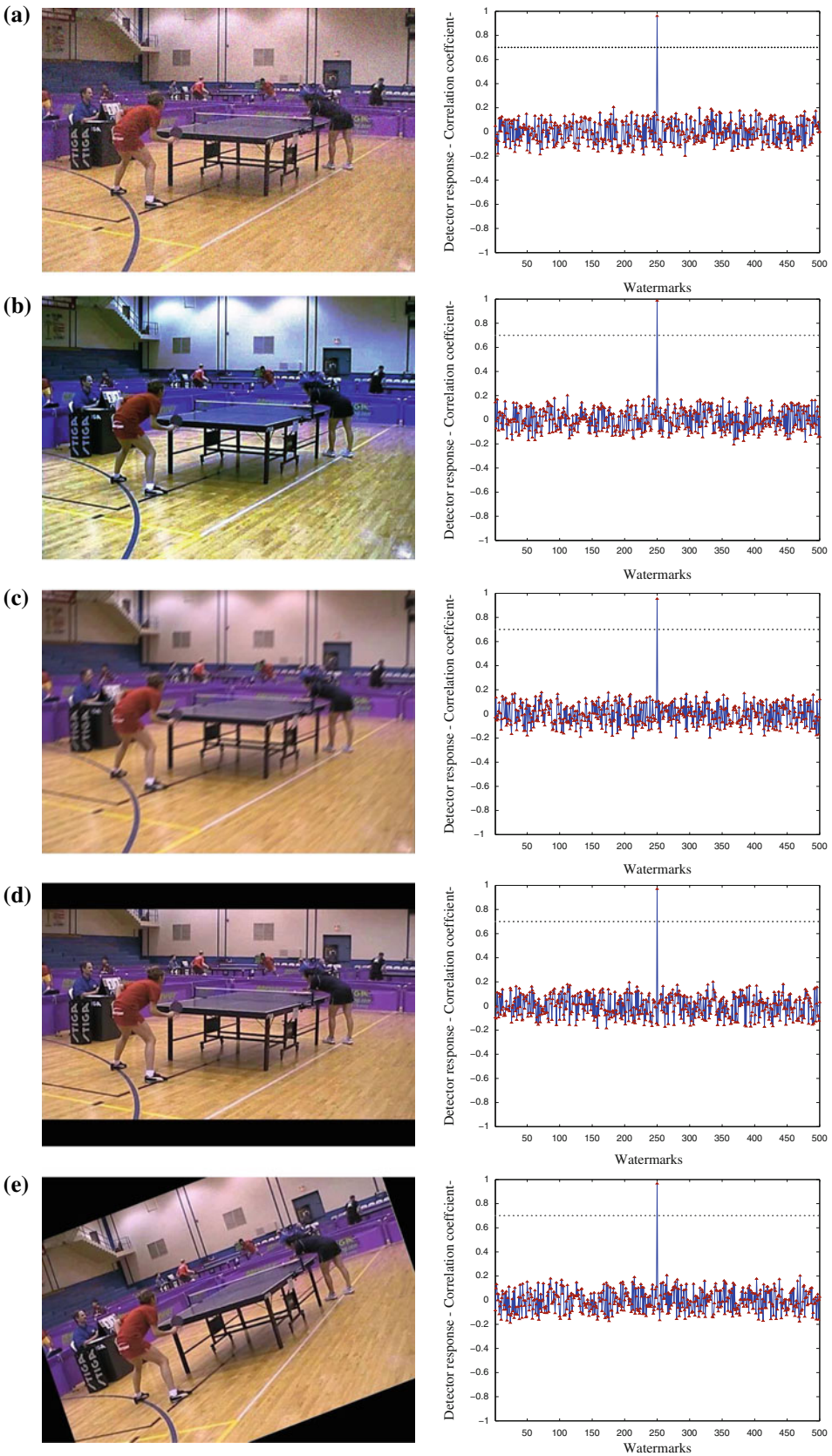

Fig. 11 Illustration of one of the MPEG tennis video I-frame under different attacks with the corresponding detector responses. The *boldface numbers* indicate the best correlation coefficient values. **a** Gaussian noise ($\sigma = 0.1$): $\varrho_{LL} = -0.8181$, $\varrho_{HL} = 0.287$, $\varrho_{LH} = 0.7168$, $\varrho_{HH} = \mathbf{9616}$. **b** Histogram equalization: $\varrho_{LL} = -0.2815$, $\varrho_{HL} = 0.8507$, $\varrho_{LH} = 0.966$, $\varrho_{HH} = \mathbf{0.987}$. **c** Motion blurring: $\varrho_{LL} = \mathbf{0.9544}$, $\varrho_{HL} = 0.8953$, $\varrho_{LH} = 0.8096$, $\varrho_{HH} = 0.7556$. **d** Cropping: $\varrho_{LL} = -0.0787$, $\varrho_{HL} = 0.9772$, $\varrho_{LH} = 0.9771$, $\varrho_{HH} = \mathbf{0.9771}$. **e** Rotation: $\varrho_{LL} = -0.3068$, $\varrho_{HL} = 0.8071$, $\varrho_{LH} = -0.8438$, $\varrho_{HH} = \mathbf{0.967}$

Table 2 MPEG tennis video under different attacks with the corresponding correlation coefficients

Attacks	ϱ_{LL}	ϱ_{HL}	ϱ_{LH}	ϱ_{HH}
Rescaling 100–50–100	**0.9939**	−0.3217	−0.0559	0.8656
Salt and peppers noise 1%	0.9853	0.8565	0.8849	**0.9748**
Gamma correction	0.6965	0.7171	0.8846	**0.9578**
Low pass filtering	**0.9671**	−0.5129	−0.9127	0.8644
JPEG compression Q = 60%	**0.9863**	0.9644	0.9555	0.9093
MPEG compression Q = 80%	**0.9756**	0.9345	0.9345	0.8956
MPEG compression Q = 50%	**0.9356**	0.9056	0.8956	0.8644
Frame dropping 50%	−0.9925	0.9108	0.9206	**0.9522**
Cropping from left 10 and right 10	−0.9674	**0.9784**	0.9723	0.9759

Boldface numbers indicate the best correlation

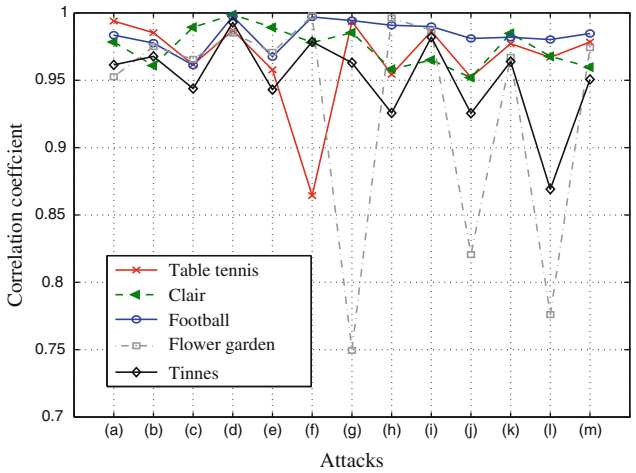

Fig. 12 Best correlation coefficient results for different video sequences. The scaling factor α used in this experiment is 0.1 for the LL band and 0.05 for all the other sub-bands. *a* Rescaling 100–50–100%, *b* Salt and peppers noise 1%, *c* Gaussian noise $\sigma = 0.2$, *d* Histogram equalization, *e* Gamma correction, *f* Low pass filtering, *g* Sharpening, *h* Motion blurring 45°, *i* JPEG compression quality = 30%, *j* Frame dropping 50%, *k* Cropping 10% from the *top* and 10% from the *bottom*, *l* Rotation 5°, and *m* Cropping 10% from the *left* and 10% from the *right*

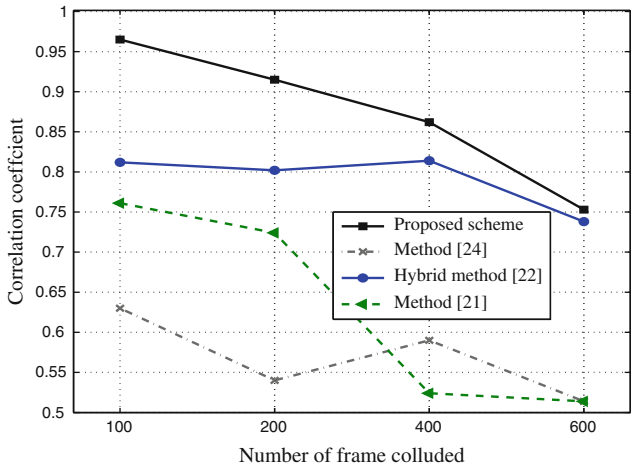

Fig. 13 Robustness against averaging attack. Comparison results between the proposed scheme and the methods introduced in [23,24,27]

In general, the accurate measurement of the perceptual quality as perceived by a human observer is a great challenge in image/video processing. The reason is that the amount and visibility of distortions introduced by the watermarking attacks strongly depend on the actual image/video content [25]. To measure the perceptual quality, we calculate the peak to signal-to-noise ratio (PSNR) that is used to estimate the quality of the watermarked frames in comparison with the original ones. The PSNR [26] is defined as follows

$$\mathrm{PSNR} = 20\log_{10}\left(\frac{\mathrm{MAX}_i}{\sqrt{\mathrm{MSE}}}\right) \tag{9}$$

where $\mathrm{MAX}_i = \max\{\widehat{F}_{ij}, 1 \le i, j \le m\}$, and the MSE is the mean squared error between the cover frame F and the watermarked frame $\widehat{F}$:

$$\mathrm{MSE} = \frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}\|F_{ij} - \widehat{F}_{ij}\|^2 \tag{10}$$

The PSNR experimental results as shown in Fig. 9, which indicate that the proposed method provides a high visual quality of the reconstructed video sequences, and hence it guarantees the watermark imperceptibility. Figure 10 shows the effect of the watermark strength factor α. Note that a smaller value of α increases the watermark imperceptibility, however it decreases the robustness against attacks.

We also performed a subjective evaluation test by involving 30 human observers (testers) who unanimously confirmed the perceptual invisibility of the hidden message. The experimental setup was carried out using four video sequences which consist of two copies for each video sequence. One copy is for the original video and another copy is for the watermarked video. During this experiment, the original and the watermarked copies were displayed on the computer screen and the testers were asked to carefully look at the

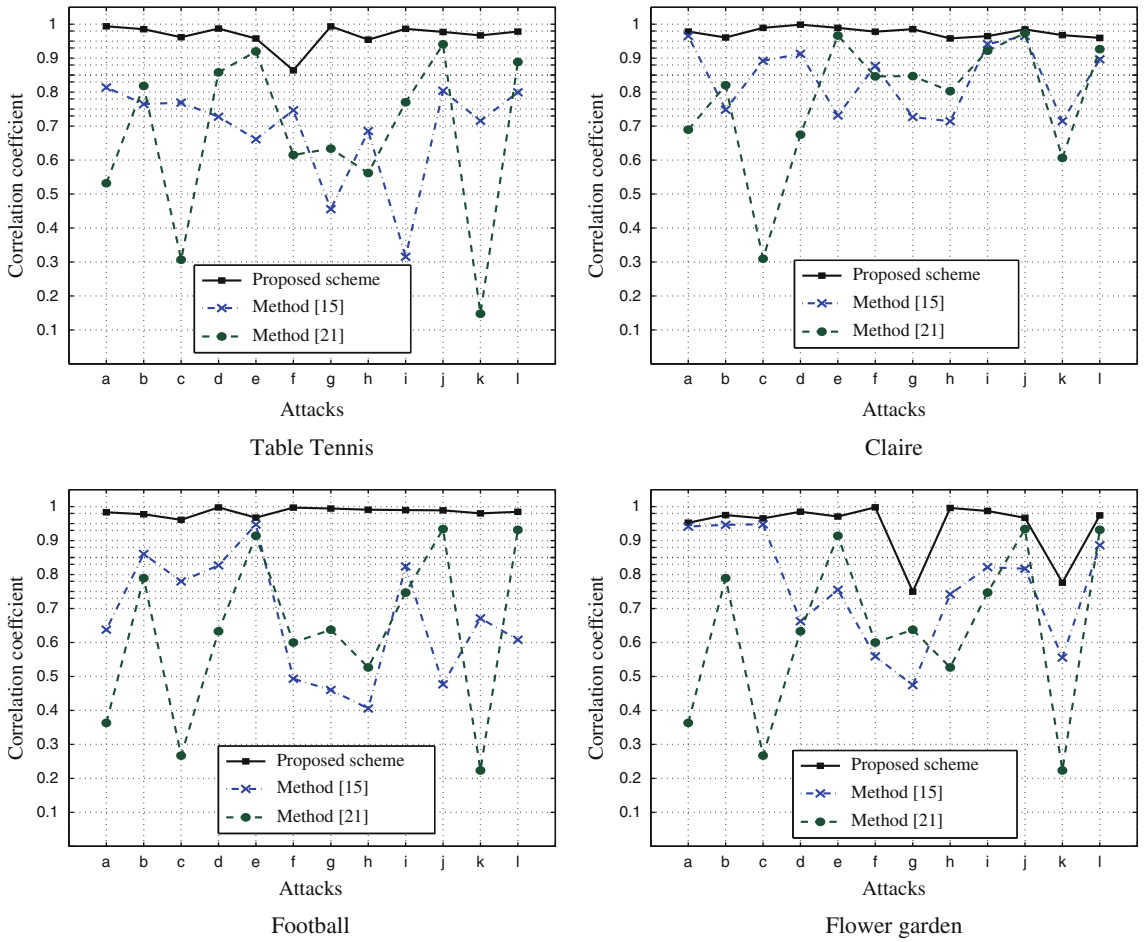

Fig. 14 Correlation coefficient comparison results between the proposed scheme and the methods introduced in [17,23]. Different video sequences distorted by different attacks are used: *a* Rescaling 100–50–100%, *b* Salt and peppers noise (2%), *c* Gaussian noise $\sigma = 0.3$, *d* Histogram equalization, *e* Gamma correction, *f* Low-pass filter (3×3), *g* Sharpening, *h* Motion blurring 45°, *i* JPEG compression quality = 25%, *j* Cropping 8% from the *top* and 8% from the *bottom*, *k* Rotation 20°, *l* Cropping 8% from the *left* and 8% from the *right*

displayed videos for a number of times, and then report if there was any visual difference between the two videos. To fully test the perceptual invisibility of the hidden message, we also repeated the experiment using different watermark strength factors. The perceptual invisibility results are shown in Table 1.

5.2 Robustness

To assess the robustness of our proposed method, we applied different attacks to the watermarked video sequence. These attacks include rescaling, rotation, Gaussian noise, histogram equalization, gamma correction, low-pass filtering, sharpening, motion blurring, frame compression, frame dropping, frame swapping, frame averaging, cropping, and also combinations of these attacks. For these attacks, we display one of the attacked I-frames and the best detector response for the real watermark, as well as 499 randomly generated other watermarks. For all the detector responses, the correlation coefficient between the original watermark and the extracted watermark is located at 250 on the X-axis. The gray dotted line at 0.7 on the Y-axis represents the threshold. This threshold is chosen experimentally to decrease false-positive alarm (presenting incorrectly the watermark in a video) and false-negative alarm (failing to detect the watermarked Video). If the correlation is larger than 0.7, then the watermark is present. Figure 11 shows one of the watermarked frames with different kinds of attacks and their corresponding best extracted watermarks. For each attack, we extracted four watermarks from the four tensors, and then we selected the best watermark that has the highest correlation coefficient with the original watermark. The caption of each sub-figure of Fig. 11 displays the correlation coefficient between the original and the four extracted watermarks. The boldface numbers indicate the best correlation. Table 2 displays the results for some other attacks. Figure 12 illustrates the robustness of our proposed method against attacks for different video sequences.

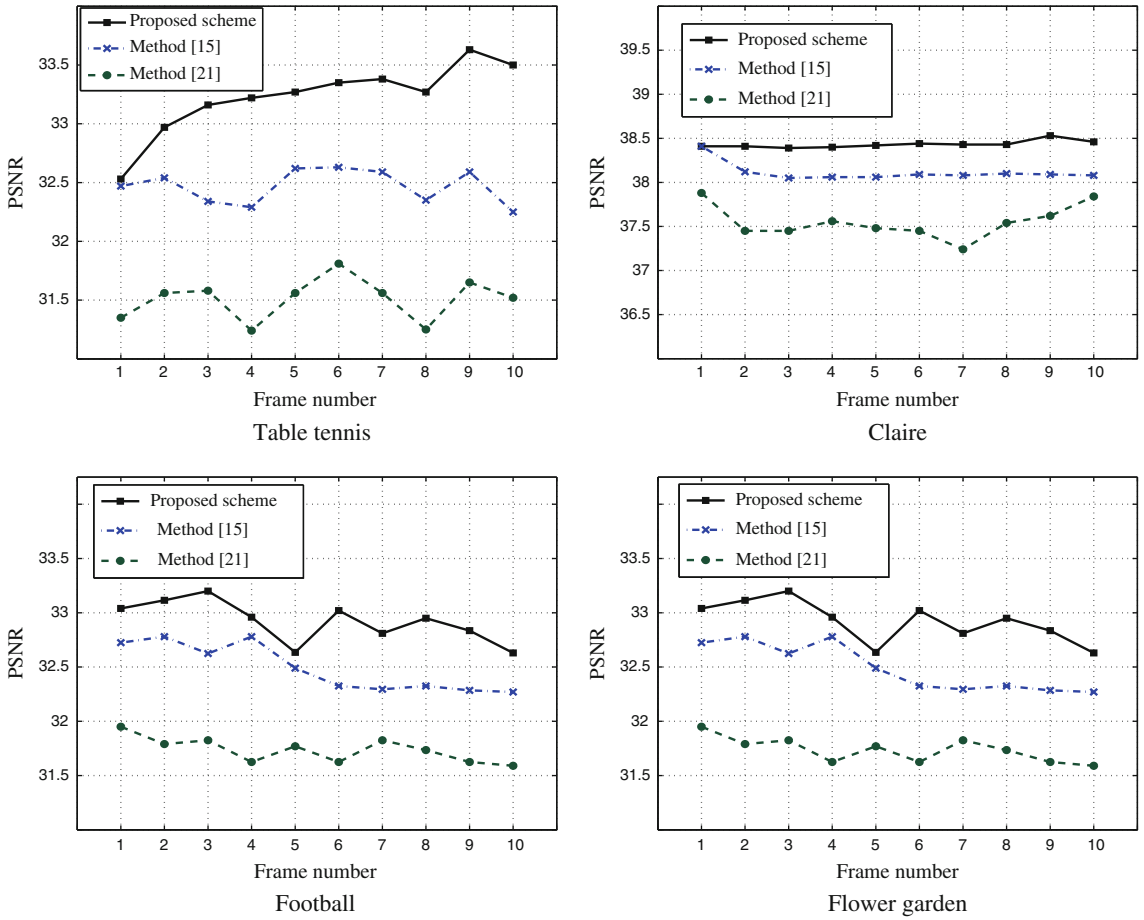

Fig. 15 PSNR comparison results between the proposed scheme and the method introduced in [17,23]. Ten successive frames are chosen from one video scene of each video sequence

The results obtained from our experiments clearly indicate the robustness of the proposed algorithm against the commonly used attacks in videos.

5.2.1 Frame dropping

Any video sequence may contain a large number of redundancies between the frames. So, the frame drooping attack is very common and effective on video watermarking. The watermark is embedded into the frames of a scene, and due to the large amount of redundancies between frames, the calculated SVD for the 3D tensor will not change significantly by frame dropping up to 60% of the highly correlated frames. To test the performance of the proposed method against the frame dropping attack, we dropped different percentages of the video frames and then we obtained the correlation coefficients between the original watermark and the extracted watermark. As shown in Fig. 16a, the proposed method achieves better performance as compared to other methods. Similar results were obtained under frame swapping attack.

5.2.2 Frame averaging

Frame averaging is another common attack in video watermarking. The attackers can use multiple frames and try to eliminate the watermark by statistical averaging of the watermarked video frames [8]. In the proposed algorithm we used different watermarks for each scene. This can prevent attackers from colluding with frames from completely different scenes to extract the watermark. Also, we used the same watermark within the same scene in order to prevent the attackers from statistically compare and remove the watermark from the motionless regions in the successive video frames. A video sequence of 8 scenes and 1,100 frames was used to test the performance of the proposed method against this attack. Figure 13 shows the robustness of the proposed method against frame averaging attack.

5.2.3 Scaling

Geometric transformations are the simplest attacks used to test the watermark detectors. Scaling is one of the very

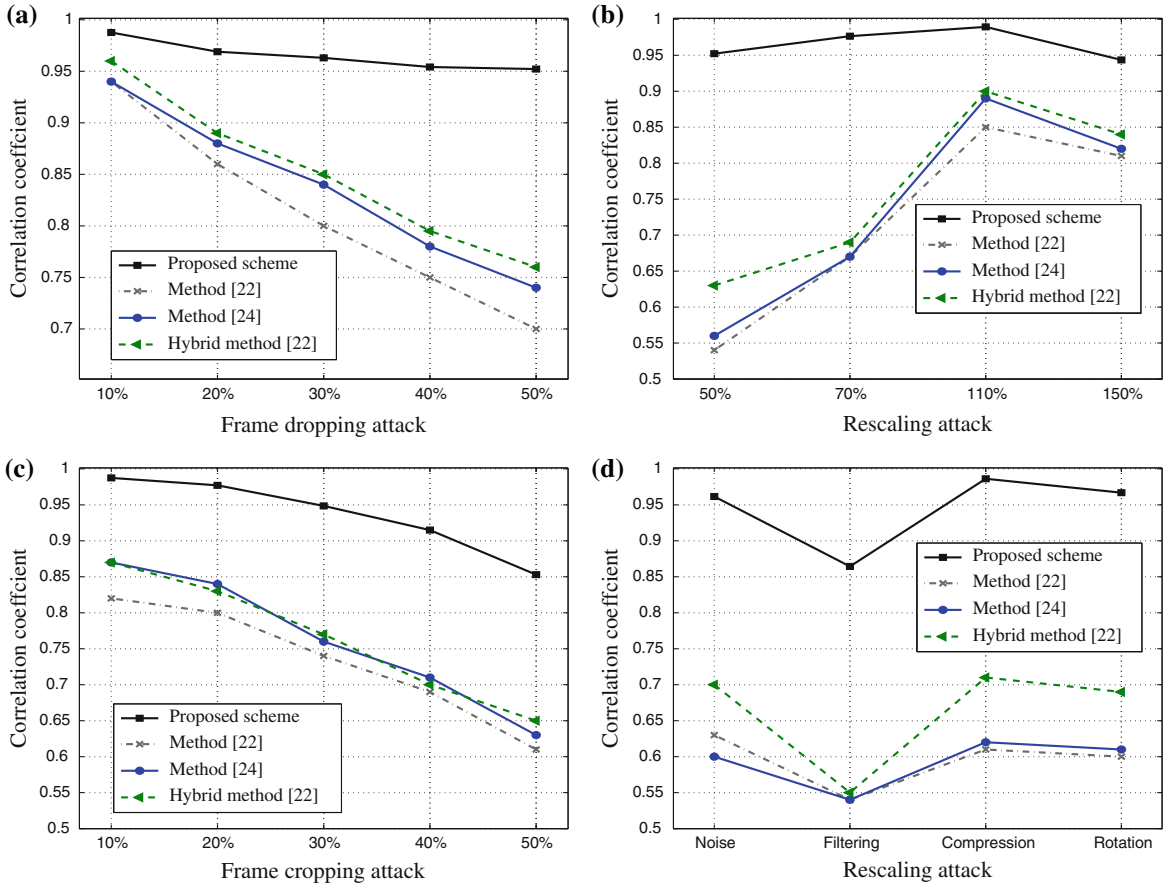

Fig. 16 Comparison results between the proposed scheme and the methods introduced in [24,27]

common geometric attacks in video watermarking. To investigate the robustness of the proposed method against this attack, we applied the scaling operation with factors of 50, 70, 110, and 150% on the watermarked video frames. Figure 12 shows the robustness of the proposed scheme against scaling attack for different video sequences, and Fig. 16b depicts the correlation coefficients for different watermarking schemes under the scaling attack.

5.3 Comparison with existing techniques

We also conducted several experiments to compare the robustness of the proposed method with existing techniques. Two types of comparisons are performed: in the first set we compare the proposed technique with two non-blind methods that were proposed in Ref. [17,23]. In the second set, we compare our results with three blind techniques that were proposed in Ref. [11,24,27].

5.3.1 Comparison with non-blind techniques

We compared our scheme with the watermarking methods proposed in Ref. [17,23]. Figure 14 depicts the correlation coefficient comparisons between our proposed watermarking scheme and [17,23] under different attacks. In these comparisons, we used the table tennis, flower garden, Claire, and football as cover video sequences. The results obtained for all the attacks clearly indicate that our proposed method performs the best in terms of robustness against the attacks. We also compared the video quality of our proposed scheme with the methods in Ref. [17,23]. The same four video sequences are examined and the results are shown in Fig. 15. Clearly the proposed method gives a high visual quality of the reconstructed video, and hence it guarantees the watermark imperceptibility.

5.3.2 Comparison with blind techniques

Comparing non-blind with blind techniques is not really a fair comparison. However, we have performed some experiments to compare the proposed scheme with three blind watermarking schemes proposed in Ref. [11,24,27]. These experiments are done to indicate that the high robustness of the proposed method compared to the blind methods may help to overcome the limitations of the non-blind approaches. Figure 16a show the robustness against frame dropping attack, while Fig. 16b

depicts the robustness against rescaling attack. Figure 16c shows the result against cropping attack. Figure 16d tests the robustness against noise, rotation, compression, and low-pass filtering attacks. The results shown in Fig. 16 clearly indicate that our proposed watermarking scheme performs the best in terms of robustness against attacks.

6 Conclusions

In this paper, we introduced a simple and computationally inexpensive watermarking methodology for embedding a watermark in the transform domain of an MPEG video. The proposed watermarking scheme is based on the concepts of TSVD and DWT. The key idea is to encode a vector of random numbers into all the frequencies of the video scenes. This was carried out by modifying the highest singular values of the 3D tensors computed form the four wavelet sub-bands of the video frames. The performance of the proposed method was evaluated through extensive experiments that clearly showed a better visual imperceptibility and an excellent resiliency against a wide range of attacks.

References

1. Cox, I.J., Miller, M.L., Bloom, J.A.: Digital Watermarking. Morgan Kaufmann, San Francisco (2001)
2. Hartung, F., Kutter, M.: Multimedia watermarking techniques. Proc. IEEE **87**(7), 1079–1107 (1999)
3. Memon, N., Wong, P.: Digital watermarks: protecting multimedia content. Comm. ACM **47**(7), 35–43 (1998)
4. Bors, G., Pitas, I.: Image watermarking using block site selection and DCT domain constraints. Opt. Express **3**(12), 512–522 (1998)
5. Podilchuk, C.I., Zeng, W.J.: Image-adaptive watermarking using visual models. IEEE J. Sel. Areas Commun. **16**(4), 525–539 (1998)
6. Langelaar, G.C., Setyawan, I., Lagendijk, R.L.: Watermarking digital image and video data. A state-of-the-art overview. IEEE Signal Process. Mag. **17**(5), 20–46 (2000)
7. Gwenael, A.D., Dugelay, J.L.: A guide tour of video watermarking. Signal Process Image Commun. **18**(4), 263–282 (2003)
8. Swanson, M.D., Zhu, B., Tewfik, A.H.: Multiresolution scene-based video watermarking using perceptual models. IEEE J. Sel Areas Commun. **16**(4), 540–550 (1998)
9. Bhattacharya, S., Chattopadhyay, T., Pal, A.: A survey on different video watermarking techniques and comparative analysis with reference to H.264/AVC. In: Proceedings of the IEEE International Symposium on Consumer Electronics, pp. 1–6 (2006)
10. Liu, H., Chang, L.: Real time digital video watermarking for digital rights management via modification of VLCS. Proc. Int. Conf. Parallel Distrib. Syst. **2**, 295–299 (2005)
11. Wang, Y., Pearmain, A.: Blind MPEG-2 video watermarking robust against geometric attacks: a set of approaches in DCT domain. IEEE Trans. Image Process. **15**(6), 1536–1543 (2006)
12. Deguillaume, F., Csurka, G., ÒRuanaidh, J., Pun, T.: Robust 3D DFT video watermarking. Proc. Secur. Watermarking Mult. Content **3657**, 113–124 (1999)
13. Hartung, F., Girod, B.: Watermarking of uncompressed and compressed video. IEEE Trans. Signal Process. **66**(3), 283–301 (1998)
14. Lin, Y.R., Hsu, W.H.: An embedded watermark technique in video for copyright protection. Proc. Int. Conf. Pattern Recog. **4**, 795–798 (2006)
15. Liu, R., Tan, T.: A SVD-based watermarking scheme for protecting rightful ownership. IEEE Trans. Multimedia **4**(1), 121–128 (2002)
16. Ganic, E., Eskicioglu, A.M.: Robust DWT-SVD domain image watermarking: embedding data in all frequencies. In: Proceedings of the ACM Multimedia and Security Workshop, pp. 166–174 (2004)
17. Abdallah, E.E., Ben Hamza, A., Bhattacharya, P.: MPEG video watermarking using tensor singular value decomposition. In: Proceeding of the International Conference on Image Analysis and Recognition. Lecture Notes in Computer Science, vol. 4633, pp. 772–783 (2007)
18. Mallat, S.: A Wavelet Tour of Signal Processing. Academic, San Diego (1998)
19. Lathauwer, L.D., Moor, B.D., Vandewalle, J.: A multilinear singular value decomposition. SIAM J. Matrix Anal. Appl. **21**(4), 1253–1278 (2000)
20. Park, S.W., Savvides, M.: Individual kernel tensor-subspaces for robust face recognition: a computationally efficient tensor framework without requiring mode factorizatin. IEEE Trans. Syst. Man Cybern. **37**(5), 1156–1166 (2007)
21. Bader, B.W., Kolda, T.G.: MATLAB tensor classes for fast algorithm prototyping. ACM Trans. Math. Softw. **32**(4), 635–653 (2006)
22. Kolda, T.G.: MATLAB Tensor Toolbox. Version 2.2. http://csmr.ca.sandia.gov/tgkolda/TensorToolbox/ (2007)
23. Elbasia, E., Eskicioglu, A.M.M.: MPEG-1 video semi-blind watermarking algorithm in the DWT domain. In: Proceedings of the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, pp. 87–90. Las Vegas (2006)
24. Chan, P.W., Lyu, M.R., Chin, R.T.: A novel scheme for hybrid digital video watermarking: approach evaluation and experimentation. IEEE Trans. Circuits Syst. Video Technol. **15**(12), 1638–1649 (2005)
25. Winkler, S., Drelie Gelasca, E., Ebrahimi, T.: Toward perceptual metrics for video watermark evaluation. Proc. SPIE Appl. Digit. Image Process. **5203**, 371–378 (2003)
26. Netravali, A.N., Haskell, B.G.: Digital Pictures: Representation, Compression, and Standards. Plenum, New York (1995)
27. Niu X., Sun S. (2000) A new wavelet-based digital watermarking for video. In: Proceedings of the IEEE Digital Signal Processing Workshop, pp. 1–6. Texas (2000)