Blind Camcording-Resistant Video Watermarking in the DTCWT and SVD Domain

MD. ASIKUZZAMAN1, HANNES MAREEN2, (Member, IEEE), NOUR MOUSTAFA1, (Senior Member, IEEE), KIM-KWANG RAYMOND CHOO3,4, (Senior Member, IEEE), AND MARK R. PICKERING1, (Member, IEEE)

1School of Engineering and Information Technology, UNSW Canberra Cyber, The University of New South Wales, Canberra, ACT 2610, Australia
2IDLab, imec, Department of Electronics and Information Systems, Ghent University, 9000 Ghent, Belgium
3Department of Information Systems and Cyber Security, The University of Texas at San Antonio, San Antonio, TX 78249, USA
4UniSA STEM, University of South Australia, Adelaide, SA 5095, Australia

Corresponding author: Hannes Mareen (hannes.mareen@ugent.be)

This work was supported in part by the Research Foundation—Flanders (FWO) under Grant 1S55218N.

ABSTRACT Video watermarking techniques can be used to prevent unauthorized users from illegally distributing videos across (social) media networks. However, current watermarking solutions are unable to embed a perceptually invisible watermark which is robust to the distortions introduced by camcording. These watermark-disrupting distortions include lossy compression, the addition of noise, frame-rate conversion and geometric distortions. In this paper, we present a novel video watermarking technique that is blind and robust to camcording attacks. The proposed approach uses the integration of the dual-tree complex wavelet transform (DTCWT) and singular value decomposition (SVD) to achieve robustness against geometric attacks. The experimental results validate our technique’s superior imperceptibility and robustness to several attacks when compared to existing peer mechanisms. In conclusion, the proposed technique can be used to protect against illegal distribution of video content.

INDEX TERMS Video watermarking, singular value decomposition (SVD), dual-tree complex wavelet transform (DTCWT), geometric attacks, camcording attacks.

I. INTRODUCTION

Illegal video redistribution by digital pirates causes financial harm to the original copyright owners or producers of films and television series [1]. For example, approximately $1.37 billion was lost to the Australian economy due to movie theft in 2010 [2]. Consequently, the security of video applications against digital piracy has become one of the most important issues for both the industry and the research community.

Digital watermarking has been broadly used for a variety of applications, including the tracking of digital pirates, preservation of copyright and playback control [3]–[9]. This paper focuses on the latter. For example, an Internet gateway could scan for the presence of a watermark and filter user’s requests accordingly. That is, a user’s request for a video downloaded can be cancelled if a watermark is detected, or the request can be responded to if no watermark is detected. For such applications, robust watermarking is required, meaning that the watermarks can survive signal processing attacks. This is in contrast to fragile watermarks that are often used for data-integrity and tamper-detection applications, which should not survive attacks, but instead manipulations can be found using the destroyed watermark locations.

The development of watermarking schemes that are robust to common attacks has remained a significant challenge to overcome [6], [10]. For example, attacks that cause de-synchronization between a watermark encoder and decoder such as camcording attacks are easily performed by digital pirates. Due to these attacks, existing blind watermark decoders are either completely unable to extract the watermark, or can only detect it with a large error [8]. Blind watermarking means that the original (unwatermarked) video is not required during watermark detection, nor any other information extracted from the original video [3].

In conventional watermarking techniques [11]–[16], the watermark decoders either require the original video for correct detection, or fail to detect the watermark when a com-
bination of temporal synchronization, signal processing and geometric attacks is applied. They have suffered from frame-rate conversion, a very popular signal processing attack which causes temporal distortion. Additionally, they suffer from camcording which causes a combination of temporal, geometric and color distortions. Techniques using the complex wavelet transform (CWT) can overcome the limitations of lack of shift invariance and improper directional selectivity by including limited redundancy in the transform, but cannot achieve efficient reconstruction higher than level 1. The use of the dual-tree complex wavelet transform (DTCWT) can overcome these limitations as it is approximately shift invariant [17]. In addition, the use of singular value decomposition (SVD) can improve the stability and performance of the DTCWT because small perturbations in the spatial domain do not change them significantly [8], [18].

In our previous work [19], we proposed a watermarking approach in the SVD and DTCWT domain. If all of the frames in a video sequence are temporally synchronized, this scheme achieves robustness to geometric attacks, H.264/AVC compression and noise addition. However, it fails to detect the watermark when a temporal synchronization attack such as frame dropping, frame insertion or frame averaging is applied. As a result, it cannot tackle frame-rate conversion and camcording. In this study, a novel video watermarking technique is proposed, which is an extension of our preliminary work and is robust to such types of temporal synchronization attacks.

The proposed technique is developed by integrating the SVD and DTCWT approaches. The main contributions of this study are provided below:

- The proposed technique is designed using the integration of SVD and DTCWT techniques applied to a chrominance (U) component of the video to achieve imperceptibility of the watermark and to prevent geometric distortion attacks.

- The proposed watermark extraction is blind because it does not require the original video, neither any other information extracted from the original video (such as the original SVs). This is in contrast to conventional SVD-based schemes.

- The extraction is robust against temporal attacks such as frame-rate conversion and camcording. That is because the extraction of the watermark from a frame depends only on that frame rather than that of multiple frames of a video sequence.

- The imperceptibility is thoroughly evaluated and compared to the state of the art, using both a subjective and objective quality assessment.

- The robustness performance of the proposed technique is thoroughly assessed. These experiments revealed that our method has a much better performance compared to classical watermark algorithms.

- We also analyze the security of the embedded watermark against a multiple watermark embedding attack.

The remainder of the paper is arranged as follows. Section II discusses the related studies of the proposed technique. A brief overview of the DTCWT and SVD is presented in Section III. Section IV explains the proposed watermarking technique. A detailed analysis of the results is discussed in Section V. Finally, the study is concluded in Section VI.

II. RELATED WORK

There have been multiple digital watermarking techniques developed in the literature which use the SVD domain to embed the watermark [7], [8], [10], [18]–[22]. For example, the authors of [21] altered the SVs of an image with the watermark. They then applied the SVD on the modified watermark again to obtain new SVs. The watermarked image was then found by substituting the original SVs with the new ones, i.e., the method is not blind. The watermark extraction from the distorted version of the watermarked image was achieved by performing the reverse operation at the decoder. The outcomes revealed that this approach was robust to JPEG compression, filtering, rotation, scaling and cropping. The authors in [7] suggested a similar watermarking technique using the SVD where the watermark was inserted directly into the SVs and extracted using the reverse operation.

Lai et al. suggested a mechanism using the SVD and discrete wavelet transform (DWT) [8]. In this mechanism, the SVD was performed on two sub-bands of a 1-level DWT decomposition of the watermark and original image. Then, the SVs of these sub-bands of the original image were modified by the SVs of the same sub-bands of the watermark. In the algorithm suggested by Makbol et al. [23], a redundancy DWT (RDWT) was used with the SVD for embedding the watermark. The watermark was embedded into the 4 sub-bands of a 1-level RDWT decomposition and then an inverse RDWT was executed to provide the watermarked image. Then, every sub-band was used to elicit the watermark. Similarly, Prasetyo et al. [22] proposed an SVD-based watermarking scheme. More specifically, the LL-band of a DWT-transformed host frame is divided in non-overlapping blocks, and the principal components of the scrambled watermark signal is embedded into the largest singular value of each block. For watermark extraction, the largest singular values of the blocks are extracted and compared to the original singular values. For both the method of Makbol et al. and the method of Prasetyo et al., the original SVs are needed to extract the watermark at the decoder, and hence the availability of the host image at the decoder was necessary. This means that these techniques are not suitable for many applications where it is not feasible to have the host image available at the decoder.

Several image and video watermarking techniques have been suggested for various applications [3], [24]–[36]. The techniques that transact with geometric distortions can be decomposed into feature-, synchronization- and invariant transform-based algorithms. In the feature-based approaches [37]–[39], the watermark is embedded into the geometric and invariant features of a video frame. The main
limitation of this type of approaches is false feature points detection [40]. These are detected wrongly when a geometric attack is applied and as a result, a false detection of the watermark is produced. In addition, these techniques are used for image rather than video watermarking applications as it is hard to obtain the same salient feature points in every frame of a video sequence.

Synchronization-based techniques validate geometric distortions before detecting the watermark. These techniques estimate the geometric parameters based on a holistic search, template addition or image registration. After that, the original image format is recovered using these parameters to elicit the watermark from the rectified image. The watermark decoder synchronizes the watermark by finding its spatial position through a comprehensive search. It is worth mentioning that this requires high computational resources and raises the probability of false detection while searching in a large space [41]. Image registration algorithms address the issue by using the watermark with a reference registration model before extracting the watermark [42]–[44]. This technique is exploited to restore the transformation parameters in the geometrically distorted version of the watermarked frame. Registration techniques are efficient for non-blind or supervised watermarking mechanisms. For example, Li et al. [25] utilize the scale invariant feature transform (SIFT) to restore geometrically attacks, in combination with a watermark in the contourlet domain. However, this synchronization requires the original video during watermark extraction. Watermark detection in blind or unsupervised watermarking systems is a much more difficult problem to solve. Template addition algorithms have also been used to secure image and video systems against geometric attacks [45], [46]. In these algorithms, a template is added in the watermark embedding procedure. The template does not convey any information but is utilized at the decoder to identify the transformation parameters before extracting the watermark. The major drawback of these algorithms is the ability of a hacker to identify and remove the template by, for example, deleting the peak components in the discrete Fourier transform (DFT) domain [38].

Invariant transform watermarking algorithms exploit the advantage of the embedding domain’s invariance to geometric distortions. In current state-of-the-art approaches, the majority of the techniques [47]–[49] use the Fourier-Mellin Transform (FMT) algorithm which provides rotation- and scale-invariant characteristics that are robust to attacks involving rotation, scaling and translation (RST). Despite FMT-based algorithms being efficient in theory, they are not applicable for real-time systems as they require a large amount of computational processing and are also not resilient to cropping [44], [50]. Another interesting approach is using the polar harmonic transform (PHT). For example, Xu [51] proposed a rotation and scale invariant image watermarking method based on the PHT, but it is not robust to cropping.

Loo et al. [52] proposed an alternative scheme using the DTCWT technique that has proper directional selectivity and approximate shift invariance characteristics that provides inherent robustness to geometric distortions [17]. As a result, other researchers have also adopted this domain for embedding the watermark [53]–[56]. In [54], the level 3 and level 4 components of a 4-level DTCWT decomposition are used to embed the watermark based on a spread spectrum mechanism. However, this mechanism does not support blind detection. In [55], the watermark was inserted into the magnitude of the highest two levels of a 4-level DTCWT decomposition. This scheme was shown to be effective in the presence of upscaling, cropping, rotation and lossy compression. However, as the human visual system (HVS) more easily perceives changes in luminance than in chrominance [57], the performance was limited by the low watermark magnitude required to maintain the imperceptibility of the watermark. Therefore, when required to maintain watermark imperceptibility, its robustness to attacks was shown to be lower than that of methods where the watermark was embedded in the chrominance channel [58], [59].

In the literature, a combination of the DTCWT and SVD has also been used to design image and video watermarking algorithms [19], [60]–[62]. Abdallah et al. [60] described an algorithm using the SVD and DTCWT domains in which the SVs of the level 2 sub-bands of a 2-level DTCWT decomposition of a video frame are used to embed the watermark. Although the watermark decoder does not require the original video to extract the watermark, the SVs of the unwatermarked video are required, making the method only semi-blind. The outcomes of the assessment of this technique showed that its robustness performance against signal processing attacks was superior to other DWT-SVD based techniques. However, the durability of the watermark in the presence of geometric distortion attacks was not considered. Another scheme proposed in [19] also used the DTCWT-SVD domain for embedding the watermark. However, this algorithm was unable to cope with temporal synchronization attacks. In other words, the watermark detection using this method failed in the presence of frame dropping, frame insertion or frame averaging attacks. In [61], Yadav et al. proposed an image watermarking algorithm using a combination of the DTCWT, principal component analysis (PCA) and SVD domain. They obtained the score matrix of the low-frequency DCTWT coefficients using the PCA. Then the SVD was applied on the score matrix to get their SVs. Finally, the resultant DTCWT-PCA-SVD features are combined with the same features extracted from the watermark to generate a watermarked image. Although this method achieved robustness against signal processing and cropping attacks, the availability of the original SVs was required at the decoder to extract the watermark, which limits the watermarking applications. In [62], a video watermarking technique based on the finite state machine was proposed in the DTCWT-SVD domain. In this method, the watermark was generated by the finite state machine, which was then embedded into the SVs of the low-frequency DTCWT coefficients of the video frames. The watermark at the decoder was extracted according to the predefined relationship of the SVs. Experimental results show the algorithm was robust against
transcoding, noise addition and temporal synchronization attacks, however, it was unable to survive geometric attacks.

More traditional transforms in image and video watermarking are the DFT and discrete cosine transform (DCT). For example, Sun et al. [63] utilize the DFT domain to provide a geometrically robust algorithm. Although the performance is good and the method is blind, it cannot cope with temporal attacks. Additionally, the DCT-based approach suggested in [59] utilizes the low-frequency DCT coefficients of a frame to embed the watermark. As a modification of the DC coefficient creates temporal flickering, this region is avoided for embedding the watermark. However, there is still slight flickering in the watermarked video because of the modification of the large coefficients around the DC component in the chrominance channel. Other techniques e.g., [64], [65], also embed the watermark using the DCT. In [64], the authors exploited the DCT coefficients to embed the watermark and showed that the watermark was robust to a downsampling attack but could not achieve robustness against rotation, upscaling and cropping. In [65], a video watermarking technique was introduced based on the DCT domain where watermark minimal sequences (WMSs) were used for embedding the watermark. Although the watermark in this scheme claims to be robust against geometric and temporal attacks, it has a limited defense against cropping and rotation. It could detect the watermark in a frame-dropping attack if at least one WMS was present at the decoder. However, as frame-rate change results in no WMS being present, it could not survive this type of attack.

Table 1 summarizes the discussed related methods. In summary, the state-of-the-art techniques are unable to fulfill the main requirements of the digital video watermarking: blind detection, robustness against signal processing, geometric and temporal synchronization attacks, and imperceptibility of the embedded watermark. As a solution, the proposed method overcomes the above challenges, i.e., it detects the imperceptible watermark without any reference or original video content, and is robust against a large variety of commonly-used attacks.

### III. OVERVIEW OF THE DTCWT AND SVD

This section discusses the background of the DTCWT and SVD which are utilized to develop the proposed technique.

#### A. DUAL-TREE COMPLEX WAVELET TRANSFORM (DTCWT)

The DTCWT was suggested by Nick Kingsbury to solve the challenges of shift-invariance and poor directional selectivity of the traditional wavelet transform [17]. The DTCWT has two trees where one provides the real portion and the other the imaginary portion of the wavelet coefficients. This approach is responsible for the high performance of the DTCWT compared with the DWT and CWT that employ a single filter tree to produce the wavelet coefficients. The decomposition structure of the DTCWT is shown in Fig. 1. It possesses the characteristics of suitable directional selectivity, perfect reconstruction, approximate shift-invariance and effective order-N estimation. Shift invariance is approximately estimated by doubling the sampling probabilities in Tree A and
Tree B through eliminating downsampling by 2 after the level 1 filters $H_0$, $H_1$, $G_0$ and $G_1$. In order for the samples at this level to be evenly spaced, the delays of $H_0$ and $H_1$ are one sample offset from those of $G_0$ and $G_1$. This shift-invariance property can be used when developing a video watermarking mechanism, which is resistant to rotation and scaling.

The DTCWT technique has a redundancy of 4:1 for 2D signals. Each level of a 2D DWT produces three sub-bands that are estimated at angles of 0°, 45° and 90° whereas a 2D DTCWT generates six subbands at angles of ±15°, ±45° and ±75°. Fig. 2(a) shows each level with six directional sub-bands of a three-level DTCWT, with the magnitudes of the identical sub-band coefficients of the Lena image shown in Fig. 2(b). This redundancy in the DTCWT plays a key role in generating durable watermarks. It should also be noted that when a random watermark is added directly to the coefficients in a redundant domain, some of its components may be lost when the inverse operation (DTCWT) is performed [14]. Therefore, we consider the DTCWT coefficients of both the video frame and watermark in our proposed embedding process.

**A. CREATION OF THE WATERMARK PATTERN**

A watermark is an identifiable pattern embedded in original video content, which could be a logo, signature, image or any other type of content. In our technique, the watermark $w \in \{-1, +1\}$ is pseudo-randomly generated pattern using a key $K$, which is exploited to create a unique pattern $w$ for $C$ consecutive frames. We select the optimal length of $C$ experimentally, since it is a trade-off for robustness against temporal frame averaging (TFA) and watermark estimation re-modulation (WER) attacks [66]. Note that, even though the pseudo-random watermark pattern changes every $C$, the proposed method is robust against temporal attacks since the detection does not require the watermark pattern.

The SVs of the transform (DTCWT) coefficients of $w$ are embedded in the host sequence, as further described in Section IV-B. In order to do this, the level 1 coefficients, $H_{1,i}^w$, of a one-level DTCWT decomposition of $w$ are selected. These coefficients are defined as

$$H_{1,i}^w = \begin{bmatrix} H_{1,i}^w(1, 1) & H_{1,i}^w(1, 2) & \cdots & H_{1,i}^w(1, M) \\ H_{1,i}^w(2, 1) & H_{1,i}^w(2, 2) & \cdots & H_{1,i}^w(2, M) \\ \vdots & \vdots & \ddots & \vdots \\ H_{1,i}^w(M, 1) & H_{1,i}^w(M, 2) & \cdots & H_{1,i}^w(M, M) \end{bmatrix}$$

where $i = 1, 2, \cdots, 6$ indicate the directional sub-bands of the complex coefficients at angles of ±15°, ±45° and ±75°. The size of $H_{1,i}^w$ in a certain sub-band is $M \times M$ which are the orthogonal (or unitary) matrices, and

$$S = \begin{bmatrix} s_1 & 0 & \cdots & 0 \\ 0 & s_2 & 0 & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & s_N \end{bmatrix}$$

is a diagonal matrix. $s_1, s_2, \cdots, s_N$, of $S$ are the diagonal elements of $S$ called the SVs of $f$. The SVs illustrate intrinsic algebraic characteristics of an image [21]. The SVD technique is utilized in video watermarking because the good stability characteristic of its SVs provides robustness to attacks.

**B. SINGULAR VALUE DECOMPOSITION (SVD)**

Let $f$ denote one frame of a video sequence. If $f$ has a squared matrix $N \times N$, the SVD of $f$ is declared by

$$f = USV^T$$

such that

$$U = \begin{bmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,N} \\ u_{2,1} & u_{2,2} & \cdots & u_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ u_{N,1} & u_{N,2} & \cdots & u_{N,N} \end{bmatrix}$$

and

$$V = \begin{bmatrix} v_{1,1} & v_{1,2} & \cdots & v_{1,N} \\ v_{2,1} & v_{2,2} & \cdots & v_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ v_{N,1} & v_{N,2} & \cdots & v_{N,N} \end{bmatrix}$$
is 8 times lower than that of the frame’s untransformed U channel. As the SVs are exploited for embedding, we apply the SVD on $H_i^u$, which is expressed as

$$H_i^u = U_i^w S_i^w (V_i^w)^T$$  \hspace{1cm} (6)

where the diagonal matrix, $S_i^w$, is defined as

$$S_i^w = \begin{bmatrix}
    s_{1,1}^w & 0 & \cdots & 0 \\
    0 & s_{1,2}^w & \cdots & 0 \\
    \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & \cdots & s_{M,1}^w \\
\end{bmatrix}$$  \hspace{1cm} (7)

The diagonal elements, $s_{1,1}^w, s_{1,2}^w, \cdots, s_{M,1}^w$, in descending order in $S_i^w$, are the SVs of $H_i^u$. The SVs of a frame are modified by these SVs based on the sign of the information bit, $b$, as discussed in Section IV-B.

### B. WATERMARK EMBEDDING

A block diagram that describes the proposed watermark creation and embedding technique is shown in Fig. 3. In short, the watermark is added in the SVs of the highest level coefficients, $H_i^u$, of a three-level DTCWT of the U frame, $f$. The highest level coefficients, i.e., low-frequency coefficients, are robust to compression and geometric distortions but have a greater influence on the perceptual quality of the video [55]. Therefore, selecting the low-frequency coefficients is a trade-off between visual quality and durability. For this reason, only the level-3 coefficients are selected for adding the watermark. The level 3 coefficients, $H_{3,i}^u$, are defined as

$$H_{3,i}^u = \begin{bmatrix}
    H_{3,i}^u(1, 1) & H_{3,i}^u(1, 2) & \cdots & H_{3,i}^u(1, M) \\
    H_{3,i}^u(2, 1) & H_{3,i}^u(2, 2) & \cdots & H_{3,i}^u(2, M) \\
    \vdots & \vdots & \ddots & \vdots \\
    H_{3,i}^u(M, 1) & H_{3,i}^u(M, 2) & \cdots & H_{3,i}^u(M, M) \\
\end{bmatrix}$$  \hspace{1cm} (8)
and the SVD of $H_{3,i}^u$ as

$$H_{3,i}^u = U_{3,i}^u S_{3,i}^u (V_{3,i}^u)^T$$

(9)

where the diagonal matrix, $S_{3,i}^u$, is defined by

$$S_{3,i}^u = \begin{bmatrix} s_{3,1} & 0 & \cdots & 0 \\ s_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & s_{3,M} \end{bmatrix}$$

(10)

If we group the diagonal matrix, $S_{3,i}^u$, into three pairs, $(S_{3,1}^u, S_{3,2}^u, S_{3,3}^u)$, $(S_{3,2}^u, S_{3,3}^u, S_{3,4}^u)$, and $(S_{3,3}^u, S_{3,4}^u, S_{3,5}^u)$. Fig. 4(a) shows that, in an unwatermarked frame, the difference between the mean of the SVs of each pair is very small. For example, the corresponding means in the figure of both $S_{3,1}^u$ and $S_{3,6}^u$ are approx. 35, the means of both $S_{3,2}^u$ and $S_{3,5}^u$ are approx. 20, and the means of both $S_{3,3}^u$ and $S_{3,4}^u$ are approx. 43. Since the means of these pairs are approximately equal in unwatermarked frames, the main goal of our watermarking method is to create a sufficiently-large difference between them.

More specifically, the method modifies two of the three pairs, namely $(S_{3,1}^u, S_{3,2}^u, S_{3,3}^u)$ and $(S_{3,2}^u, S_{3,3}^u, S_{3,5}^u)$. That is because modifications in those bands affect the resulting watermarked frame less than modifications of the other two sub-bands. This is shown in Fig. 4(b), which shows the average PSNR of the transformed version of the watermark obtained from Eq. (7). The SVs of $S_{3,i}^u$ are modified by the corresponding SVs of $S_{3,i}^w$ as follows

$$\tilde{S}_{3,i}^u = \begin{cases} S_{3,i}^u + b\beta S_{3,1}^w, & \text{for } i = 1, 2 \\ S_{3,i}^u - b\beta S_{3,1}^w, & \text{for } i = 5, 6 \\ S_{3,i}^u, & \text{otherwise} \end{cases}$$

(11)

where $\beta$ controls the strength of the watermark and $b \in [-1, +1]$ is the embedding bit pattern. Note that the value of $\beta$ is directly proportional to the watermark’s robustness and is inversely proportional to its transparency. Hence the selected value of $\beta$ is a trade-off between the video quality and robustness against attacks. At this stage of the embedding process, the diagonal matrix, $S_{3,i}^u$, in Eq. (9) is replaced by the modified diagonal matrix, $\tilde{S}_{3,i}^u$, in Eq. (11). The watermarked level 3 coefficients, $\tilde{H}_{3,i}^u$, are given by

$$\tilde{H}_{3,i}^u = U_{3,i}^u \tilde{S}_{3,i}^u (V_{3,i}^u)^T$$

(12)

Finally, a watermarked video frame, $\tilde{f}$, is generated by taking an inverse transform of the modified DTCWT coefficients. The overall watermark embedding process for a video frame is summarized in Algorithm 1. This process is repeated for each frame in a video sequence.

**C. WATERMARK EXTRACTION**

After embedding a watermark in a video sequence, it can be compressed for storage purposes or subjected to geometric and/or signal processing attacks. After subjecting a watermarked frame, $f$, to attacks, the attacked frame is denoted as $\tilde{f}$. An overall block diagram of the extraction phase is given in Fig. 5. Firstly, a three-level DTCWT is applied to the U frame of $\tilde{f}$. Although the watermark embedding process modified the SVs of the level 3 coefficients, it is extracted from the SVs of any level (1, 2 or 3) to avoid a
downscaling in resolution attack. That is because a watermark can be extracted from a lower DTCWT level if the video was
downscaled, as examined in previous work [58].

The SVD of the level-\(l\) complex coefficients of the U frame of \(\hat{f}\) is given by

\[
H^{u}_{i,l} = U^{u}_{i,l} \tilde{S}^{u}_{i,l} (V^{u}_{i,l})^{T} \quad \text{for } i = 1, 2, 5, 6
\]

where \(l = 1, 2\) or 3 are the DTCWT decomposition levels. The SVs of the diagonal matrix, \(\tilde{S}^{u}_{i,l}\), are defined as

\[
\tilde{S}^{u}_{i,l} = \begin{bmatrix}
S^{u,1}_{l,i} & 0 & \cdots & 0 \\
0 & S^{u,2}_{l,i} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & \cdots & \cdots & S^{u,M}_{l,i}
\end{bmatrix} \quad \text{for } i = 1, 2, 5, 6
\]

Before these values were modified during watermark embedding, the difference between the means of the unwatermarked SVs of a pair is negligible. However, the watermark encoder created a large difference between them, using Eq. (11). The mean difference using sub-bands \(i = 1\) and 6 is expressed as

\[
D_{1} = \frac{1}{M} \sum_{j=1}^{M} \tilde{S}^{u,j}_{1,1} - \frac{1}{M} \sum_{j=1}^{M} \tilde{S}^{u,j}_{1,6}
\]

and, using sub-bands \(i = 2\) and 5, is given by

\[
D_{2} = \frac{1}{M} \sum_{j=1}^{M} \tilde{S}^{u,j}_{2,2} - \frac{1}{M} \sum_{j=1}^{M} \tilde{S}^{u,j}_{2,5}
\]

Since we supplied the same \(b\) at the encoder to add the watermark in both pairs of sub-bands, the sign of both \(D_{1}\) and \(D_{2}\) should be the same, i.e., either both positive or both negative. Hence, the embedded bits can be extracted based on the signs of \(D_{1}\) and \(D_{2}\). More specifically, the bits are respectively estimated from \(D_{1}\) and \(D_{2}\) by

\[
b_{1} = \begin{cases}
1, & \text{if } D_{1} > 0 \\
-1, & \text{otherwise}
\end{cases}
\]

and

\[
b_{2} = \begin{cases}
1, & \text{if } D_{2} > 0 \\
-1, & \text{otherwise}
\end{cases}
\]

Using these equations, we obtained \(b_{1}\) and \(b_{2}\) for a single frame. We further utilize the notations \(b_{1}(k)\) and \(b_{2}(k)\), which denote \(b_{1}\) and \(b_{2}\) decoded from the \(k\)th frame in a video. After decoding the bits from \(P\) consecutive frames of a sequence, we apply \(b_{1}(k) \ast b_{2}(k)\), where the \(\ast\) symbol represents a normalized cross-correlation (NCC) between \(b_{1}(k)\) and \(b_{2}(k)\). As we embed the same \(b\) in both pairs, the NCC should provide a large correlation peak. In contrast, when no watermark was embedded, the NCC should be approximately zero.

---

**Algorithm 1 Watermark Embedding in the U Frame of a Video Sequence**

**Require:** \(w\): Watermark, \(f\): Video frame, \(b\): Information bit, \(\beta\): Watermark embedding strength

1. Apply a one-level DTCWT on \(w\) to obtain the level 1 complex coefficients, \(H^{w}_{1,i}\).
2. Apply a three-level DTCWT on the U frame of \(f\) to get coefficients, \(H^{u}_{3,i}\).
3. for \(i := 1\) to 6 do
   4. Compute the SVD of \(H^{w}_{3,i} = U^{w}_{3,i} S^{w}_{3,i} (V^{w}_{3,i})^{T}\)
   5. Compute the SVD of \(H^{u}_{3,i} = U^{u}_{3,i} S^{u}_{3,i} (V^{u}_{3,i})^{T}\)
   6. if \(i = 1\) or 2 then
      7. \(\tilde{S}^{u}_{3,i} = S^{u}_{3,i} + \beta S^{w}_{3,i}\)
   8. else if \(i = 3\) or 4 then
      9. \(\tilde{S}^{u}_{3,i} = S^{u}_{3,i}\)
   10. else if \(i = 5\) or 6 then
       11. \(\tilde{S}^{u}_{3,i} = S^{u}_{3,i} - \beta S^{w}_{3,i}\)
   12. end if
4. end for
5. Update the level 3 coefficients, \(\hat{H}^{u}_{3,i} = U^{u}_{3,i} \tilde{S}^{u}_{3,i} (V^{u}_{3,i})^{T}\)
6. Return the watermarked frame, \(\hat{f}\), by taking the inverse
DTCWT of \(\hat{H}^{u}_{3,i}\)
Algorithm 2 Watermark Extraction From a Video Sequence

Require: \( \tilde{F} \): Watermarked video subjected to attacks, 
\( P \): Number of frames, \( Th \): Threshold

1: for \( k := 1 \) to \( P \) do
2: Apply a three-level DTCWT on the U component of the \( k^{th} \) frame of \( \tilde{F}, \tilde{f} \), to get the complex coefficients, \( H^u_i \)
3: for \( l := 1 \) or 2 or 3 do
4: for \( i := 1, 2, 5 \) and 6 do
5: Estimate the SVD of \( \tilde{H}^u_i \) \( = \tilde{U}^u_i \tilde{S}^u_i (\tilde{V}^u_i)^T \)
6: Find the SVs of \( \tilde{S}^u_i \) using (14)
7: end for
8: Estimate \( D_1 \) and \( D_2 \) using (15) and (16) respectively
9: if \( D_1 > 0 \) then
10: \( b_1(k) = 1 \)
11: else
12: \( b_1(k) = -1 \)
13: end if
14: if \( D_2 > 0 \) then
15: \( b_2(k) = 1 \)
16: else
17: \( b_2(k) = -1 \)
18: end if
19: end for
20: end for
21: if \( b_1(k) \cdot b_2(k) > Th \) then
22: Watermark present
23: else
24: Watermark absent
25: end if

Finally, we compare the peak of the correlation output with the threshold, \( Th \) (see Section V-C1), to examine if the watermark is present or not. The above process is summarized in Algorithm 2.

As shown in Algorithm 1 and Algorithm 2, the proposed method can only embed a single information bit in the video. Hence, the embedding capacity is limited to only a single bit, regardless of the robustness and imperceptibility performance. Still, this information bit can be used to flag protected media. For example, the watermark extraction filter can be placed in an Internet gateway to scan the existence of the watermark. Then, a user’s request for a video downloaded can be cancelled if a watermark is detected, or the request can be responded to if no watermark is detected. Future work can investigate how to blindly extract a larger payload in the SVs.

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

This section evaluates and discusses the proposed watermarking method in comparison with the state of the art. First, Section V-A describes the experimental setup. Then, Section V-B, V-C, and V-D discuss the imperceptibility, robustness, and security, respectively. Finally, Section V-E evaluates the computational complexity.

A. EXPERIMENTAL SETUP

In this part of our study, comprehensive experiments were carried out to evaluate the performance of the proposed technique. In order to justify its performance, ten publicly-available standard test sequences, BasketBallDrive, BQTerrace, Cactus, IntoTree, OldTownCross, ParkJoy, Life, ControlledBurn, SpeedBag and PedestrianArea [67], [68], with HD resolutions of 1080 × 1920, were adopted for our experiments. The key, \( K \), was used to create \( w \) for \( C = 7 \) consecutive frames. The DTCWT decomposition level for extracting the watermark was selected using the resolution of the input video at the decoder. It is level 3 (\( l = 3 \)) for a resolution of 1080 × 1920, level 2 (\( l = 2 \)) for the resolutions of 540 × 960, 480 × 640 and 270 × 480, and level 1 (\( l = 1 \)) for a resolution of 240 × 320. The NCC was performed after decoding the bits of \( P = 300 \) consecutive frames of a watermarked content and the level of robustness was assessed based on the false negative rate (FNR) of the NCC peak of these decoded bit patterns.

The effectiveness of the proposed technique was compared with three schemes: the DWT-SVD method by Prasetyo et al. [22], the DCT method by Lee et al. [64], and the DCT method by Ling et al. [65]. As summarized in Table 1, these methods claim to have robustness against signal processing attacks and (some) geometric attacks. Moreover, the DCT-based methods claim robustness against temporal attacks. Note that the method by Prasetyo et al. was adapted such that the watermark is embedded into every frame rather than in key frames only (as is done in all other evaluated methods), and no image scrambling was applied on the watermark signal since the signal is random already. Furthermore, a watermark signal size of 480 × 270 and a scaling factor \( \alpha \) of 0.1 was used, which is the same as in their originally proposed method. Furthermore, Lee et al. justified the performance of their method using bit error rates. The bit pattern was extracted from each frame using this algorithm and all 0 bits were set to \(-1\). Finally, to compare the FNR of this method with our algorithm, we performed the NCC between this pattern and the embedded one.

B. WATERMARK IMPERCEPTIBILITY

Because of the non-linear property of the HVS, objective quality measures may not correspond to a real output of the visual quality of video content. On the other hand, subjective quality assessments, which are based on human judgement, also vary from person to person. Therefore, in this paper, both subjective and objective assessments were conducted to reflect the quality of the watermarked sequences.

1) SUBJECTIVE QUALITY ASSESSMENT

In this work, a subjective assessment was carried out based on the double-stimulus continuous quality scale (DSCQS) approach specified in the ITU-R standard [69], using five
test sequences: ParkJoy, Life, ControlledBurn, SpeedBag and PedestrianArea. The subjective tests were utilized to examine the strength of the imperceptible watermark, $\beta$, in Eq. (11) and to evaluate the quality of a watermarked video. We also performed the subjective tests for Ling’s and Lee’s methods. In order to do this, we embedded watermarks in the previously mentioned sequences using five different Q-step sizes, $\Delta = 150, 250, 350, 450$, and $547$, embedding strength ratios, 0.01, 0.02, 0.03, 0.04, and 0.05 and watermark strengths, $\beta = 36, 38, 40, 42$, and $44$ for Lee’s, Ling’s and the proposed algorithms, respectively. The resulting 75 watermarked sequences (i.e., 5 sequences $\times$ 15 embedding parameters) were then judged by 15 participants in three small groups. It should be noted that the participants, both male and female, were postgraduate students at the University of New South Wales, Canberra, Australia. Four of them were researching in the area of image processing and others were from different fields. Based on the DSCQS method, we simultaneously displayed a pair of sequences, one was the original and the other one watermarked, on a 60-inch television. Their positions were set randomly and the people were unaware of which the area of image processing and others were from different fields. Based on the DSCQS method, we simultaneously displayed a pair of sequences, one was the original and the other one watermarked, on a 60-inch television. Their positions were set randomly and the people were unaware of which the watermarked sequence. An assessment sheet with a continuous scale (see Fig. 6) was provided to each participant.) for each embedding parameter of each method were obtained. Their mean opinion scores (MOSs) are plotted in Fig. 7, where the error bars indicate a 95% confidence interval. The mean of the scores for the original sequences is also depicted by green horizontal lines. Although Ling et al. [65] and Lee et al. [64] recommended a strength ratio of 0.05 and $\Delta = 547$, respectively, Fig. 7(a) and Fig. 7(b) show that the watermarked video exhibited highly perceptible distortions using the recommended parameters and the MOSs obtained from these algorithms were well below those of the original sequence. On the other hand, Fig. 7(c) shows that the MOSs were close to the average score of the unwatermarked video sequences for the proposed scheme. Although the MOS for each $\beta$ is very close to the green line, $\beta = 38$ (from Eqn. (11)) was selected for embedding the watermark using the proposed method.

![FIGURE 6. Scale to evaluate the quality of the original and watermarked videos.](image)

**TABLE 2.** Objective quality measure of the watermarked video in terms of the PSNR, SSIM, VIFP and VMAF.

| Method     | PSNR (dB) | SSIM (%) | VIFP (%) | VMAF (%) |
|------------|-----------|----------|----------|----------|
| Prasetyo [22] | 44.626    | 99.120   | 92.063   | 98.645   |
| Lee [64]    | 46.123    | 99.820   | 94.770   | 94.255   |
| Ling [65]   | 30.398    | 98.884   | 97.982   | 97.896   |
| Proposed    | 56.916    | 99.919   | 98.217   | 99.076   |

### 2) OBJECTIVE QUALITY ASSESSMENT

The imperceptibility of the watermarked video was objectively evaluated using the peak signal-to-noise ratio (PSNR), structural similarity (SSIM) [70], pixel-based visual information fidelity (VIFP) [71] and video multimethod assessment fusion (VMAF) [72]. In order to average the obtained values of the Y, U, and V channel, a weighted average methodology was used [73]. The average quality values of the watermarked video sequences using Prasetyo’s, Lee’s, Ling’s and the proposed methods are summarized in Table 2. As we embedded the watermark into the U channel, this table shows that the quality of the watermarked video using the proposed method imperceptible, and is better in terms of the PSNR, SSIM, VIFP, and VMAF compared to the state-of-the-art techniques.

### C. ROBUSTNESS TO ATTACKS

In this part of the experimental analysis, we analyzed the robustness of our proposed algorithm and state-of-the-art methods to commonly-used attacks. That is, we embedded twenty unique patterns of the watermark in each video sequence. The FNR was then computed by fitting a Gaussian distribution to the NCC peaks which were obtained using these watermark patterns.

#### 1) PROBABILITY OF FALSE DETECTION

The correlation results for the extracted bits were compared with a threshold to determine whether the video was watermarked. In order to do this, we defined the threshold for providing a probability of false detection, $P_{fd}$, as $10^{-6}$. This probability is estimated by the model described in [74], and is computed by

$$P_{fd} = \int_0^{\cos^{-1}(Th)} \sin^{n-2}(x) dx = \frac{\int_0^{\pi/2} \sin^{n-2}(x) dx}{\sin^n(0) - 1}$$

where $n$ is the length of the extracted bit pattern and $Th$ the watermark detection threshold. Prasetyo’s, Ling’s and Lee’s schemes also used this formula to calculate the thresholds, $Th$. These were computed numerically as 0.0132, 0.4265, 0.6308 and 0.2700 for Prasetyo’s, Ling’s, Lee’s and the proposed algorithms, respectively.
2) DOWNSCALING IN RESOLUTION AND ASPECT-RATIO CHANGE

In this section, we discuss the performance of the proposed technique for downscaling to arbitrary resolutions and aspect-ratio changes. The FNRs of Prasetyo’s, Ling’s, Lee’s and the proposed methods when evaluating without attack and downscaling to the resolutions of $540 \times 960$, $480 \times 640$, $270 \times 480$ and $240 \times 320$ are shown in Table 3. These were zero for each scheme except Prasetyo’s algorithm, which reports a FNR of 0.488% when downscaled to a resolution of $240 \times 320$. The resolution of the original video was $1080 \times 1920$, which corresponds to the aspect ratio of 16:9. However, as the aspect ratios of the downscaled sequences were 16:9 and 4:3, and the FNRs were still zero at 4:3, we deduced that the watermark of the proposed scheme was robust to aspect-ratio change.

### TABLE 3. False negative rates (%) for downscaling in resolution.

| Downscaling to       | Prasetyo [22] | Lee [64] | Ling [65] | Proposed |
|----------------------|---------------|----------|-----------|----------|
| Original Resolution  | 0.000         | 0.000    | 0.000     | 0.000    |
| $540 \times 960$     | 0.000         | 0.000    | 0.000     | 0.000    |
| $480 \times 640$     | 0.000         | 0.000    | 0.000     | 0.000    |
| $270 \times 480$     | 0.004         | 0.000    | 0.000     | 0.000    |
| $240 \times 320$     | 0.488         | 0.000    | 0.000     | 0.000    |

3) GEOMETRIC ATTACKS

Geometric distortions are very common types of attacks in the area of digital watermarking and consist of upscaling and downscaling in resolution, cropping, rotation and aspect-ratio change. In our experiment for an upscaling attack, firstly, we scaled up each sequence to a certain level (1% to 15%) and then cropped to the original dimension. Finally, the resultant sequences were additionally downsampled to resolutions of $270 \times 480$ and $240 \times 320$. The performances of all approaches are shown in Table 4. It is clear that Prasetyo’s method extracted the watermark with a very high false-negative error, and Lee’s method could only withstand a upscaling-and-cropping attack of 1% but failed for stronger attacks. Note that ‘–’ indicates that the watermark detection failed. Although Ling’s approach performed well at relatively low levels of upscaling and cropping, it was still inadequate at higher levels. On the contrary, our proposed scheme extracted the watermark with zero FNRs for up to 15% of upscaling and cropping, and even when additionally downsampled to the dimension of $240 \times 320$.

To evaluate the robustness of our proposed approach to a rotation attack, each sequence was rotated at various angles (between 1 and 15 degrees) and then cropped to remove any newly created zero pixels from the border of the resultant frame. We also extracted the watermark from downscaled versions of the rotated sequences. The results for a combination of upscaling, cropping, rotation and resizing to $270 \times 480$ and $240 \times 320$ are shown in Table 5. These results indicate that, although Ling’s method provided low FNRs at small angles of rotation, only the proposed approach was robust to these attacks.

4) ADDITION OF WHITE GAUSSIAN NOISE

To assess robustness to the addition of noise, white Gaussian noise with zero mean and several different variances were added to the watermarked video contents. It should be noted that the pixel values of the distorted frames were constrained to the range 0 to 255. The resultant FNRs after extracting the watermark from noisy sequences are shown in Table 6. This table illustrates that the FNRs of Prasetyo’s, Lee’s, Ling’s and the proposed algorithms were zero or close to zero even when the resolution scaled down to $240 \times 320$ pixels and the PSNR of the attacked watermarked $U$ channel decreased to 20.45 dB for the proposed scheme.

5) JOINT ATTACK

In this test, a joint attack consisting of the addition of white Gaussian noise, upscaling, downscaling in resolution, rotation and cropping were considered. Firstly, we added white Gaussian noise with zero mean and a variance of 65 into each frame of the watermarked video sequences and then the noisy sequences were rotated at different angles and cropped as
explained previously. Finally, the resultant video sequences were downsampled to 270 × 480 and 240 × 320 pixels resolutions before passing through the decoder. The performances of our proposed method shown in Table 7 indicate that it was far better for the joint attack than those of Prasetyo’s, Ling’s and Lee’s schemes.

6) H.264/AVC COMPRESSION

In this part of our analysis, we evaluate our technique to a lossy compression attack. A H.264/AVC encoder compressed the sequences using the quantization parameter (QP) of 28, at 25 frames per second (fps). We applied the attack using H.264/AVC because it is the most common video encoder.

Note that we have no reason to expect different results when using other compression standards such as H.265/HEVC. Although Prasetyo’s and Lee’s schemes did not achieve zero FNRs for compression attacks, Ling’s and the proposed techniques achieved better robustness as shown in Table 8.
In the previous section, we analyzed the performances of our method in terms of downscaling in resolution, upscaling, rotation, cropping, and the addition of Gaussian noise and combinations of these attacks. However, in this experiment, we additionally analyzed the robustness to these attacks in combination with H.264/AVC compression. The FNRs of the proposed method, Prasetyo’s, Ling’s and Lee’s approaches are summarized in Table 9 to Table 12. Table 9 indicates that our proposed scheme extracted the watermark without any error for a combination of cropping, upscaling, and compression and downscaling in resolution, even for 15% upscaling and cropping. On the contrary, although the FNRs of Ling’s scheme were small for up to 5% upscaling and cropping, they were large for higher values of the upscaling attack. This table also shows that the FNRs of Prasetyo’s and Lee’s methods were very high even for a 1% upscaling attack.

The FNRs of Prasetyo’s, Ling’s and Lee’s methods were better at higher variance values than those of the proposed method, Prasetyo’s, Ling’s and Lee’s approaches failed when combined with rotation, upscaling and cropping in the presence of white Gaussian noise with zero mean and a variance of 65, and H.264/AVC compression, as shown in Table 12.

A receiver operating characteristic (ROC) curve is a trade-off between false positive rate (FPR) and true positive rate (TPR). A smaller FPR and larger TPR represents the better performance of the watermark detection. The ROC curves for Prasetyo’s, Lee’s, Ling’s and the proposed algorithms after jointly applying H.264/AVC compression, the additive Gaussian noise of variance 65, 5° rotation and cropping, and resizing the dimension to 1080×1920, 270×480 and 240×320 pixels resolutions are presented in Fig. 8. These curves indicate that our scheme is more robust to a joint attack than the other approaches, even at a resolution of 240×320 pixels.
TABLE 12. False negative rates (%) for H.264/AVC compression (QP = 28), additive Gaussian noise with variance of 65, rotation, cropping and downscaling in resolution.

| Rotation and cropping (degree) | Original Resolution | Downscaling to 270 × 480 | Downscaling to 240 × 320 |
|-------------------------------|------------------|----------------------|----------------------|
|                               | Prasetyo [22]    | Lee [64] | Ling [65] | Proposed | Prasetyo [22]    | Lee [64] | Ling [65] | Proposed | Prasetyo [22]    | Lee [64] | Ling [65] | Proposed |
| 1                             | 99.979          | 99.999   | 0.090    | 0.000    | 99.987         | 99.999   | 0.063    | 0.000    | 99.998         | 99.999   | 0.078    | 0.000    |
| 3                             | 99.999          | 100.000 | 39.666   | 0.000    | 99.999         | 100.000 | 39.167   | 0.000    | 99.999         | 100.000 | 39.455   | 0.000    |
| 5                             | 99.999          | 100.000 | 66.999   | 0.000    | 99.999         | 100.000 | 69.425   | 0.000    | 99.999         | 100.000 | 77.998   | 0.001    |
| 7                             | –               | –        | 82.349   | 0.005    | –               | –        | 86.692   | 0.000    | –               | –        | 93.164   | 0.804    |
| 9                             | –               | –        | 89.298   | 0.770    | –               | –        | 92.742   | 0.550    | –               | –        | 98.811   | 4.267    |
| 11                            | –               | –        | 84.596   | 4.967    | –               | –        | 97.426   | 2.409    | –               | –        | 99.570   | 5.268    |
| 13                            | –               | –        | 91.953   | 10.752   | –               | –        | 99.212   | 1.194    | –               | –        | 99.929   | 0.714    |
| 15                            | –               | –        | 98.365   | 19.223   | –               | –        | 99.893   | 0.016    | –               | –        | 99.994   | 0.000    |

FIGURE 8. ROC curves for different schemes for the joint attack which consisted of compression, Gaussian noise with variance 65, 5° rotation and cropping and downscaling to (a) 1080 × 1920, (b) 270 × 480 and (c) 240 × 320 pixels resolutions.

7) CAMCORDING AND TEMPORAL SYNCHRONIZATION ATTACKS

For the camcording experiments, we tested only Lee’s and the proposed algorithm using five different watermarked video sequences ParkJoy, Life, ControlledBurn, SpeedBag and PedestrianArea. Ling’s method was not used in this test because camcording causes temporal de-synchronization, for example, frame insertion or frame dropping, as well as geometric and color distortions. If the watermark extraction is dependent on consecutive frames, the watermark decoder will be unable to elicit the watermark when temporal de-synchronization is applied. As Ling’s approach requires at least one WMS to extract the watermark but no WMSs appeared due to the frame-rate change, this scheme was not robust to camcording. Furthermore, Prasetyo’s method was not considered for the camcording experiment because it is not robust to geometric distortions, and it is a non-blind method which requires temporal synchronization at the decoder. Note that the original method proposed to only embed the watermark in key frames. By detecting those key frames during watermark extraction and hence retrieving the original SVs for non-blind detection, the temporal synchronization issue could be solved. However, those key frames could be (un)intentionally dropped by an attacker during a camcording experiment, making watermark extraction impossible for their method.

In our test, the watermarked sequences were run repeatedly on a 60-inch television at 30 fps. We used a SONY HDR-TD30VE camcorder to capture each sequence 10 times starting from a different frame, for at least 300 frames with three different frame rates (25p, 50i and 50p). The 150 captured AVCHD (MPEG4-AVC/H.264) format sequences (i.e., 5 sequences × 10 trials × 3 frame rates) were re-compressed using the x264 encoder and then downscaled to the resolutions of 540 × 960, 480 × 640, 270 × 480 and 240 × 320. The FNRs after extracting the watermark from the resultant sequences for both Lee’s and the proposed algorithms are summarized in Table 13. It is clear that our algorithm was more robust to camcording, even when additionally compressing and downscaling the videos.

TABLE 13. False negative rates (%) for camcording.

| Frame Rate | Downscaling to | Lee [64] | Proposed |
|------------|----------------|----------|----------|
| 25p        | 1080 × 1920    | 7.66     | 0.13     |
|            | 540 × 960      | 7.72     | 0.04     |
|            | 480 × 640      | 7.78     | 0.00     |
|            | 270 × 480      | 8.15     | 0.02     |
|            | 240 × 320      | 8.46     | 0.00     |
| 50i        | 1080 × 1920    | 8.68     | 0.73     |
|            | 540 × 960      | 8.82     | 0.57     |
|            | 480 × 640      | 8.91     | 0.02     |
|            | 270 × 480      | 9.28     | 2.30     |
|            | 240 × 320      | 9.68     | 0.00     |
| 50p        | 1080 × 1920    | 10.36    | 5.53     |
|            | 540 × 960      | 10.45    | 4.82     |
|            | 480 × 640      | 10.50    | 0.00     |
|            | 270 × 480      | 10.85    | 2.04     |
|            | 240 × 320      | 11.17    | 0.00     |
D. WATERMARK SECURITY

Multiple watermark embedding is a possible attack that could be used to remove a watermark from the watermarked video sequence. To analyze the security of the watermark against this attack, four different sequences were used where each has different motion characteristic, and all contain 300 frames. In a multiple watermark embedding attack, we considered that the attackers have knowledge of the proposed watermark mechanisms but not of the original watermark, \( w \), which was embedded into the video sequence. Therefore, the attackers might add a second watermark into the watermarked video sequence to remove the effect of the first watermark. As the attackers have no idea of the embedded watermark pattern, we embedded a random watermark pattern into each video sequence.

Fig. 9 shows the mean of the SVs of the DTCWT coefficients in each direction after a multiple watermark embedding attack, for four test sequences. In each direction, for each sequence shown in Fig. 9, the first (i.e., blue) bar is the mean SV of the original unwatermarked video sequence, the second (i.e., red) bar shows the mean SV of the watermarked video sequence, and the final (i.e., orange) bar describes the mean SV of the randomly-watermarked video sequence where the second watermark was embedded into the already watermarked sequence. For each sequence, it is clear from the blue bars (i.e., the first bar in each direction, for the unwatermarked videos) that the difference between the mean of the SVs of each pair is very small, especially the mean differences \( D_1 \) and \( D_2 \). However, we created a large difference between them by embedding the watermark using (11), as shown in the second, red bars. In the last, orange bars, i.e., after embedding the second (random) watermark, although the difference between the directions in a pair is not large enough, the signs of both mean differences \( D_1 \) and \( D_2 \) are the same. Hence, the same bit patterns from both pairs will be extracted and the presence of the watermark will be detected accurately.

In order to justify the watermark detection accuracy for a multiple watermark embedding attack in terms of the FNR, we experimented using 20 different random watermark patterns for the watermark embedding strengths, \( \beta = 0, 2, \ldots, 90 \). For each strength and video sequence, a random watermark pattern was embedded into the already-watermarked sequence and repeated for other random patterns. It should be noted that there is no effect of
posed, Prasetyo’s, Lee’s and Ling’s methods are summarized algorithms for each frame of a video sequence. The average run time of the watermarking embedding and extraction and Prasetyo’s approach in Python 2.7.18. We computed and Ling’s methods were implemented in MATLAB 9.4, ning on a Windows 7 operating system. The proposed, Lee’s experiments were conducted on a computer with a 2.40 GHz Intel(R) Core(TM) i5-6300U CPU and 16 GB of RAM run-

FIGURE 10. False negative rate (FNR) and peak signal-to-noise ratio (PSNR_U) after the multiple watermark embedding attack into the U channel for different embedding strengths.

TABLE 14. Watermark embedding and extraction times per HD frame.

| Method      | Embedding Time (s) | Extraction Time (s) |
|-------------|--------------------|---------------------|
| Prasetyo [22] | 2.032              | 5.635               |
| Lee [64]    | 0.570              | 0.256               |
| Ling [65]   | 1.878              | 0.204               |
| Proposed    | 0.262              | 0.126               |

the second (random) watermark when β = 0, i.e., it does not modify the original watermarked sequence. Fig. 10 shows the FNR for different embedding strengths and corresponding average PSNR_U of the randomly watermarked U channel. In this figure, the PSNR_U at β = 0 indicates the quality of the watermarked U channel using the proposed technique before applying the multiple watermark embedding attack. It can also be seen that the PSNR_U of the randomly watermarked U channel decreases with increasing embedding strength, although the FNR of the proposed detection is zero or close to zero. Therefore, it can be evident that the proposed watermarking algorithm is secure against a multiple watermark embedding attack. The proposed technique can be employed to preserve copyright of the videos’ producers or owners. The technique can guard against illegal sharing with untrusted applications such as those used on social media networks.

E. COMPUTATIONAL EFFICIENCY

The computational complexity of the proposed algorithm was compared with Prasetyo’s, Lee’s and Ling’s methods. The experiments were conducted on a computer with a 2.40 GHz Intel(R) Core(TM) i5-6300U CPU and 16 GB of RAM running on a Windows 7 operating system. The proposed, Lee’s and Ling’s methods were implemented in MATLAB 9.4, and Prasetyo’s approach in Python 2.7.18. We computed the run time of the watermarking embedding and extraction algorithms for each frame of a video sequence. The average embedding and extraction times per frame using the proposed, Prasetyo’s, Lee’s and Ling’s methods are summarized in Table 14. This table shows that the proposed embedding and extraction algorithms are faster than the state-of-the-art algorithms.

VI. CONCLUSION

We proposed a novel video watermarking method that inherits the advantages of both SVD and DTCWT. That is, it embeds the watermark in the SVs of the DTCWT coefficients of the chrominance channel. We chose to use this channel as, for the case of watermark imperceptibility, it supports a higher strength watermark than would have been possible using alternative luminance embedding algorithms.

We examined the imperceptibility of our method by both subjective and objective quality assessments. From these experiments, we found that our method embeds an imperceptible watermark and outperforms the state of the art.

The DTCWT decomposition level for extracting the watermark was selected based on the resolution of the sequence. This approach helps to maintain robustness to aspect-ratio changes and any arbitrary downscaling in resolution. Note that the original video is not required during decoding, i.e., the proposed method is blind. The combined benefits of the SVD and DTCWT enhance the robustness of our scheme to geometric attacks such as upscaling, cropping and rotation. The effectiveness of our algorithm was experimentally validated against the addition of white Gaussian noise, H.264/AVC compression and combinations of these attacks. Finally, our proposed blind watermarking method outperforms existing techniques in robustness against frame-rate change and camcording attacks.

REFERENCES

[1] J. C. Ulin, The Business of Media Distribution: Monetizing Film, TV, and Video Content in an Online World, Evanston, IL, USA: Routledge, 2019.
[2] Economic Consequences of Movie Piracy—Australia, Austral. Fed. Against Copyright Theft, Australia, Jan. 2011.
[3] M. Asikuzzaman and M. R. Pickering, “An overview of digital video watermarking,” IEEE Trans. Circuits Syst. Video Technol., vol. 28, no. 9, pp. 2131–2153, Sep. 2018.
[4] A. A. Elrowayati, M. A. Alrshah, M. F. L. Abdullah, and R. Latip, “H2vc watermarking techniques for authentication and copyright applications: Challenges and opportunities,” IEEE Access, vol. 8, pp. 114172–114189, 2020.
[5] Y. Zhou, C. Wang, and X. Zhou, “An intra-drift-free robust watermarking algorithm in high efficiency video coding compressed domain,” IEEE Access, vol. 7, pp. 132991–133007, 2019.
[6] G. C.-W. Ting, B.-M. Goi, and S.-W. Lee, “Robustness attacks on video watermarking using singular value decomposition,” in Proc. 3rd Int. Conf. Digit. Signal Process., 2019, pp. 157–162.
[7] S. Dogan, T. Tuncer, E. Avci, and A. Gulten, “A robust color image watermarking method with singular value decomposition,” Adv. Eng. Softw., vol. 42, no. 6, pp. 336–346, Jun. 2011.
[8] C.-C. Lai and C.-C. Tsai, “Digital image watermarking using discrete wavelet transform and singular value decomposition,” IEEE Trans. Instrum. Meas., vol. 59, no. 11, pp. 3060–3063, Nov. 2010.
[9] H. Mareen, M. Courteaux, J. De Praeter, M. Asikuzzaman, G. Van Wallendael, and P. Lambert, “Rate-distortion-preserving forensic watermarking using quantization parameter variation,” IEEE Access, vol. 8, pp. 63700–63709, 2020.
[10] J. Liu, J. Huang, Y. Luo, L. Cao, S. Yang, D. Wei, and R. Zhou, “An optimized image watermarking method based on HD and SVD in DWT domain,” IEEE Access, vol. 7, pp. 80849–80860, 2019.
[11] P. W. Chan, M. R. Lyu, and R. T. Chin, “A novel scheme for hybrid dig... and experimentation,” IEEE Trans. Circuits Syst. Video Technol., vol. 15, no. 12, pp. 1638–1649, Dec. 2005.
D. Choi, H. Do, H. Choi, and T. Kim, “A blind MPEG-2 video watermarking robust to camcorder recording,” Signal Process., vol. 90, no. 4, pp. 1327–1332, 2010.

H. A. Abdallah, M. M. Hadhoud, and A. A. Shaalan, “SVD-based watermarking scheme in complex wavelet domain for color videos,” in Proc. Int. Conf. Comput. Eng. Syst., Dec. 2009, pp. 455–460.

J. Yadav and K. Sehra, “Large scale dual tree complex wavelet transform based robust features in PCA and SVD subspace for digital image watermarking,” Proc. Comput. Sci., vol. 132, pp. 863–872, 2018.

C. Guangxi, C. Ze, W. Daoshun, L. Shundong, H. Yong, and Z. Baoying, “Combined DTCWT-SVD-Based video watermarking algorithm using finite state machine,” in Proc. 11th Int. Conf. Adv. Comput. Intel. (ICACI), Jun. 2019, pp. 179–183.

X.-C. Sun, Z.-M. Lu, Z. Wang, and Y.-L. Liu, “A geometrically robust multi-bit video watermarking algorithm based on 2-D DFT,” Multimedia Tools Appl., vol. 80, no. 9, pp. 13491–13511, Apr. 2021.

M.-J. Lee, D.-H. Im, H.-Y. Lee, K.-S. Kim, and H.-K. Lee, “Real-time video watermarking system on the compressed domain for high-definition video contents: Practical issues,” Digit. Signal Process., vol. 22, no. 1, pp. 190–198, Jan. 2012.

H. Ling, L. Wang, and F. Zou, “Real-time video watermarking scheme resistant to geometric distortions,” J. Electron. Imag., vol. 20, no. 1, Jan. 2011, Art. no. 013025.

G. Doerr and J.-L. Dugelay, “Security pitfalls of frame-by-frame approaches to video watermarking,” IEEE Trans. Signal Process., vol. 52, no. 10, pp. 2955–2964, Oct. 2004.

F. Bossen, Common Test Conditions and Software Reference Configurations, document JCTVC-L1100, ITU-T Joint Collaborative Team on Video Coding, Jan. 2013.

L. Haglund, “The SVT high definition multi format test set,” Sveriges Television AB (SVT), Stockholm, Sweden, Tech. Rep., Feb. 2006. [Online]. Available: https://tech.ebu.ch/docs/hdtv/svt-multiformat-conditions-v10.pdf

Methodology for the Subjective Assessment of the Quality of Television Pictures, document R. I.-R. BT.500-11 Radiocommunication Sector, Int. Telecommun. Union, 2002.

Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600–612, Apr. 2004.

H. R. Sheikh and A. C. Bovik, “Image information and visual quality,” IEEE Trans. Image Process., vol. 15, no. 2, pp. 430–444, Feb. 2006.

Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy, and M. Manohara, “Toward a practical perceptual video quality metric,” Netflix Technol. Blog, Jun. 2016. [Online]. Available: https://netflixtechblog.com/toward-a-practical-perceptual-video-quality-metric-6532089652

T. K. Tan, R. Weerakkody, M. Mrak, N. Ramzan, V. Baroncini, J.-R. Ohm, and G. J. Sullivan, “Video quality evaluation methodology and verification testing of HEVC compression performance,” IEEE Trans. Circuits Syst. Video Technol., vol. 26, no. 1, pp. 76–90, Jan. 2016.

M. L. Miller and J. A. Bloom, “Computing the probability of false watermark detection,” in Proc. 3rd Int. Workshop Inf. Hiding, 1999, pp. 146–158.

MD. ASIKUZZAMAN received the B.Sc. degree in electronics and telecommunication engineering from the Rajshahi University of Engineering & Technology, Rajshahi, Bangladesh, in 2010, and the Ph.D. degree in electrical engineering from The University of New South Wales, Canberra, Australia, in 2015, under a very competitive University International Postgraduate Award Scholarship. He was a Research Associate with the School of Engineering and Information Technology, The University of New South Wales, from 2015 to 2019, where he was a Senior Research Associate, from 2019 to 2020. His current research interests include video watermarking, machine learning, remote sensing, computer vision, medical imaging, and video coding. He was the Technical Program Chair for the 2018 International Conference on Digital Image Computing: Techniques and Applications. He is currently serving as an Associate Editor for the IEEE Access.

HANNES MAREEN (Member, IEEE) received the M.Sc. degree in computer science engineering from Ghent University, Belgium, in 2017. From 2017 to 2021, he worked towards a Ph.D. as an SB Fellow at IDLab, Ghent University—imec, with the financial support of the Research Foundation—Flanders (FWO). In 2020, he was a Visiting Research Fellow with the School of Engineering and Information Technology, The University of New South Wales, Canberra, Australia. He has been working with the IDLab, Ghent University—imec, as a Post-doctoral Researcher, since 2021. His research interests include multimedia security and forensics and video coding and compression.

NOUR MOUSTAFA (Senior Member, IEEE) received the bachelor’s and master’s degrees in information systems from the Faculty of Computer and Information, Helwan University, Egypt, in 2009 and 2014, respectively, and the Ph.D. degree in the field of cyber security from UNSW, in 2017. He was a Post-doctoral Fellow in cyber security with The University of New South Wales (UNSW), Canberra, Australia, from June 2017 to February 2019. He is currently a Postgraduate Discipline Coordinator (Cyber) and a Senior Lecturer in cyber security and computing with the School of Engineering and Information Technology (SEIT), UNSW Canberra. His research interests include cyber security, particularly network security, host- and network–intrusion detection systems, statistics, deep learning, machine learning techniques, designing and developing threat detection and forensic mechanisms to the industry 4.0 technology for identifying malicious activities from cloud computing, fog computing, the IoT, industrial control systems over virtual machines, and physical systems.

KIM-KWANG RAYMOND CHOO (Senior Member, IEEE) received the Ph.D. degree in information security from the Queensland University of Technology, Australia, in 2006. He currently holds the Cloud Technology Endowed Professorship at The University of Texas at San Antonio (UTSA). He is the Founding Co-Editor-in-Chief of ACM Distributed Ledger Technologies: Research & Practice, and the Founding Chair of IEEE Technology and Engineering Management Society’s Technical Committee on Blockchain and Distributed Ledger Technologies. He is an ACM Distinguished Speaker and IEEE Computer Society Distinguished Visitor (2021–2023) and a Web of Science’s Highly Cited Researcher (Computer Science 2021, Cross-Field 2020). In 2015, he and his team won the Digital Forensics Research Challenge organized by Germany’s University of Erlangen-Nuremberg. He was a recipient of the 2019 IEEE Technical Committee on Scalable Computing Award for Excellence in Scalable Computing (Middle Career Researcher).

MARK R. PICKERING (Member, IEEE) was born in Biloela, Australia, in 1966. He received the B.Eng. degree from the Capricornia Institute of Advanced Education, Rockhampton, Australia, in 1988, and the M.Eng. and Ph.D. degrees from The University of New South Wales, Canberra, Australia, in 1991 and 1995, respectively, all in electrical engineering. He was a Lecturer, from 1996 to 1999, a Senior Lecturer, from 2000 to 2009, an Associate Professor, from 2010 to 2017, and a Professor, from 2018 to 2020, with the School of Electrical Engineering and Information Technology, The University of New South Wales, Canberra, where he is currently an Emeritus Professor. His research interests include video and audio coding, medical imaging, data compression, information security, data networks, and error-resilient data transmission.